

Network Working Group
Internet Draft
Intended status: Standards Track
Expires: January 2013

A. Bashandy
B. Pithawala
K. Patel
Cisco Systems
July 16, 2012

Scalable BGP FRR Protection against Edge Node Failure
draft-bashandy-bgp-edge-node-frr-03.txt

Abstract

Consider a BGP free core scenario. Suppose the edge BGP speakers PE1, PE2,..., PEn know about a prefix P/m via the external routers CE1, CE2,..., CEm. If the edge router PEi crashes or becomes totally disconnected from the core, it is desirable for a core router "P" carrying traffic to the failed edge router PEi to immediately restore traffic by re-tunneling packets originally tunneled to PEi and destined to the prefix P/m to one of the other edge routers that advertised P/m, say PEj, until BGP re-converges. In doing so, it is highly desirable to keep the core BGP-free while not imposing restrictions on external connectivity. Thus (1) a core router should not be required to learn any BGP prefix, (2) the size of the forwarding and routing tables in the core routers should be independent of the number of BGP prefixes, (3) provisioning overhead should be kept at minimum, (4) re-routing traffic without waiting for re-convergence must not cause loops, and (4) there should be no restrictions on what edge routers advertise what prefixes. For labeled prefixes, (6) the label stack on the packet must allow the repair PEj to correctly forward the packet and (7) there must not be any need to perform more than one label lookup on any edge or core router during steady state

Status of this Memo

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

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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

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

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

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

This Internet-Draft will expire on January 16, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

| | |
|------------------------------------------------------------------|----|
| 1. Introduction..... | 3 |
| 1.1. Conventions used in this document..... | 4 |
| 1.2. Terminology..... | 5 |
| 1.3. Problem definition..... | 6 |
| 2. Overview of the solution in an MPLS Core..... | 7 |
| 2.1. Control Plane operation for Automated pNH Assignment..... | 7 |
| 2.2. Control Plane operation for Configured pNH..... | 10 |
| 2.3. Forwarding behavior at Steady State (When pPE is reachable) | 11 |
| 2.4. Forwarding behavior when pPE Fails..... | 12 |
| 3. Overview of the solution in a Pure IP Core..... | 13 |
| 3.1. Control Plane operation..... | 13 |

| | |
|-------------------------------------------------------------------|----|
| 3.2. Forwarding Behavior at Steady State (while pPE is reachable) | 13 |
| 3.3. Forwarding Behavior at Failure (when pPE is not reachable) | 14 |
| 4. Example | 15 |
| 4.1. Control Plane | 16 |
| 4.2. Forwarding Plane at Steady State (When PE0 is reachable) | 16 |
| 4.3. Forwarding Plane at Failure (When PE0 is not reachable) | 17 |
| 5. Inter-operability with Existing IP FRR Mechanisms | 19 |
| 6. Security Considerations | 19 |
| 7. IANA Considerations | 19 |
| 8. Conclusions | 19 |
| 9. References | 20 |
| 9.1. Normative References | 20 |
| 9.2. Informative References | 21 |
| 10. Acknowledgments | 21 |
| Appendix A. How to protect Against Misconfigured pNH | 22 |
| Appendix B. Alternative Approach for advertising (pNH,rNH) to iPE | 23 |
| Appendix C. Modification History | 24 |
| A.1.1. Changes from Version 02 | 24 |
| A.1.2. Changes from Version 01 | 24 |

1. Introduction

In a BGP free core, where traffic is tunneled between edge routers, BGP speakers advertise reachability information about prefixes to other edge routers not to core routers. For labeled address families, namely AFI/SAFI 1/4, 2/4, 1/128, and 2/128, an edge router assigns local labels to prefixes and associates the local label with each advertised prefix such as L3VPN [10], 6PE [11], and Software [9]. Suppose that a given edge router is chosen as the best next-hop for a prefix P/m. An ingress router that receives a packet from an external router and destined to the prefix P/m "tunnels" the packet across the core to that egress router. If the prefix P/m is a labeled prefix, the ingress router pushes the label advertised by the egress router before tunneling the packet to the egress router. Upon receiving the packet from the core, the egress router takes the appropriate forwarding decision based on the content of the packet or the label pushed on the packet.

In modern networks, it is not uncommon to have a prefix reachable via multiple edge routers. One example is the best external path [8]. Another more common and widely deployed scenario is L3VPN [10] with multi-homed VPN sites. As an example, consider the L3VPN topology depicted in Figure 1.

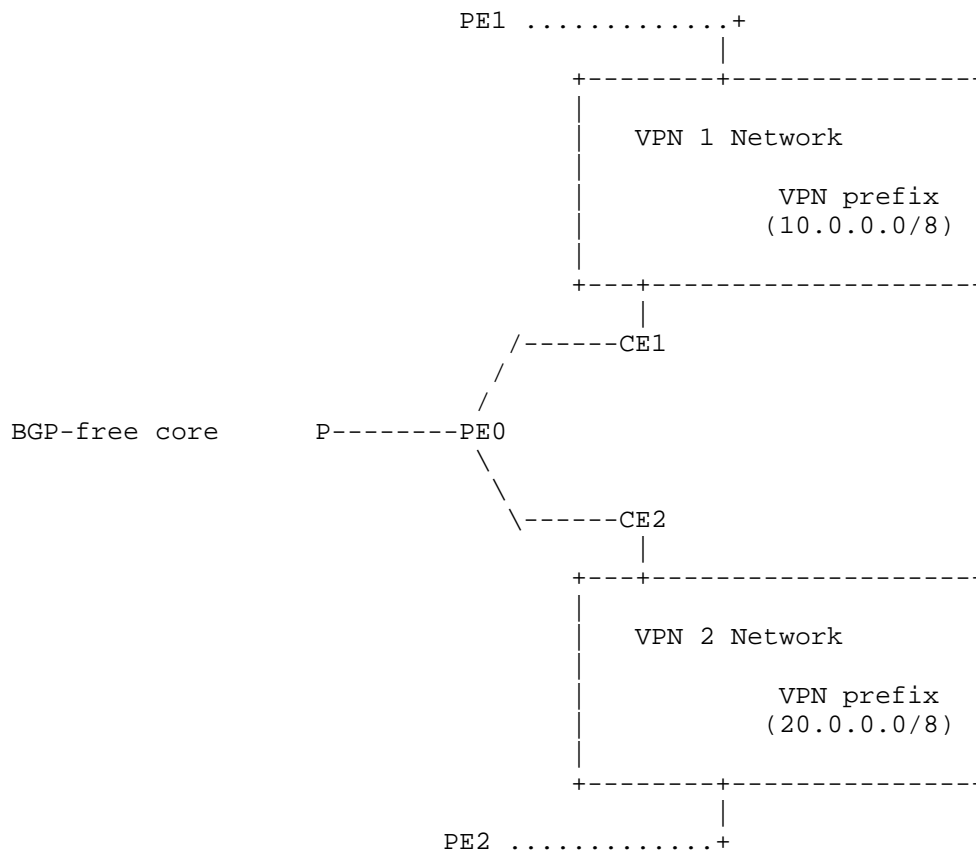


Figure 1 VPN prefix reachable via multiple PEs

As illustrated in Figure 1, the edge router PE0 is the primary NH for both 10.0.0.0/8 and 20.0.0.0/8. At the same time, both 10.0.0.0/8 and 20.0.0.0/8 are reachable through the other edge routers PE1 and PE2, respectively.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [1].

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

1.2. Terminology

This section defines the terms used in this document. For ease of use, we will use terms similar to those used by L3VPN [10]

- o BGP-Free core: A network where BGP prefixes are only known to the edge routers and traffic is tunneled between edge routers
- o External prefix: It is a prefix P/m (of any AFI/SAFI) that a BGP speaker has an external path for. The BGP speaker may learn about the prefix from an external peer through BGP, some other protocol, or manual configuration. The protected prefix is advertised to some or all of the internal peers.
- o Protectable prefix: It is an external prefix P/m of any AFI/SAFI) that a BGP speaker has an external path to and is eligible to have a repair path.
- o Primary Egress PE, "ePE": It is an IBGP peer that can reach the prefix P/m through an external path and advertised the prefix to the other IBGP peers. The primary egress PE was chosen as the best path by one or more internal peers. In other words, the primary egress PE is an egress PE that will normally be used by some ingress PEs when there is no failure. Referring to Figure 1, PE0 is an egress PE.
- o Protected egress PE, "pPE" (Protected PE for simplicity): It is an egress PE that has or eligible to have a repair path for some or all of the prefixes to which it has an external path. Referring to Figure 1, PE0 is a protected egress PE.
- o Protected edge router: Any protected egress PE.
- o Protected next-hop (pNH): It is an IPv4 or IPv6 host address belonging to the protected egress PE. Traffic tunneled to this IP address will be protected via the mechanism proposed in this document. Note that the protected next-hop MUST be different from the next-hop attribute in the BGP update message [2][3].
- o CE: It is an external router through which an egress PE can reach a prefix P/m. The routers "CE1" and "CE2" in Figure 1 are examples of such CEs.
- o Ingress PE, "iPE": It is a BGP speaker that learns about a prefix through another IBGP peer and chooses that IBGP peer as the next-hop for the prefix.

- o Repairing P router "rP" (Also "Repairing core router" and "repairing router"): A core router that attempts to restore traffic when the primary egress PE is no longer reachable without waiting for IGP or BGP to re-converge. The repairing P router restores the traffic by rerouting the traffic (through a tunnel) towards the pre-calculated repair PE when it detects that the primary egress PE is no longer reachable. Referring to Figure 1, the router "P" is the repairing P router.
- o Repair egress PE "rPE" (Repair PE for simplicity): It is an egress PE other than the primary egress PE that can reach the protected prefix P/m through an external neighbor. The repair PE is pre-calculated prior to any failure. Referring to Figure 1, PE1 is the repair PE for 10.0.0.0/8 while PE2 is the repair PE for 20.0.0.0/8.
- o Underlying Repair label (rL): The underlying repair label is the label that will be pushed so that the repair PE can forward repaired traffic correctly. A repair label is defined for labeled protected prefixes only.
- o Repair next-hop (rNH): It is an IPv4 or IPv6 host address belonging to the repair egress PE. If the protected prefix is advertised via BGP, then the repair next-hop SHOULD be the next-hop attribute in the BGP update message [2][3].
- o Repair path (Also Repair Egress Path): It is the repair next-hop. If an underlying repair label exists, the repair path is the repair next-hop together with the underlying repair label.
- o Primary tunnel: It is the tunnel from the ingress PE to the primary egress PE
- o Repair tunnel: It is the tunnel from the repairing P router to the repair egress PE

1.3. Problem definition

The problem that we are trying to solve is as follows

- o Even though multiple prefixes may share the same egress router, they have different repair edge router. In Figure 1 above, both 10.0.0.0/8 and 20.0.0.0/8 share the same primary next hop PE0, the routing protocol(s) must identify that the node protecting repair node for 10.0.0.0/8 is PE1 while the node protecting repair node for 11.0.0.0/8 is PE2

- o On loosing connection to the edge router, the core router "P" MUST reroute traffic towards the *correct* repair edge router without waiting for IGP or BGP to re-converge and update the routing tables. On the failure of PE0 illustrated in Figure 1, the core router P needs to reroute traffic for 10.0.0.0/8 towards PE1 and traffic for 11.0.0.0/8 towards PE2
- o The repairing core router P MUST NOT be forced to learn about the BGP prefixes on any of the edge router. The same applies for all core routers.
- o The size of the routing table on any core router MUST be independent of the number of BGP prefixes in the network.
- o Rerouting traffic without waiting for IGP and BGP to re-converge after a failure MUST NOT cause loops.
- o For labeled prefixes, when a packet gets re-routed to the repair PE, the label stack on the packet MUST ensure correct forwarding.
- o Provisioning overhead must be kept at minimum. In addition, misconfiguration should be detectable.
- o At steady state, when pPE is reachable, a path taken by traffic flow must not be impacted by enabling the solution proposed in this document on some or all routers

2. Overview of the solution in an MPLS Core

The solution proposed in this document relies on the collaboration of egress PE, ingress PE, penultimate hop routers, and repairing router. This section gives an overview of how the solution works for labeled and unlabeled protected prefixes in an MPLS core.

2.1. Control Plane operation for Automated pNH Assignment

This section outlines the solution for the case where the protected next hop "pNH" is automatically calculated instead of being assigned by an operator.

1. Each egress router that is capable of handling repaired traffic assigns each protectable labeled prefix a repair label: "rL". "rL" is advertised as optional path attribute. "rL" MUST be per-CE or per-VRF for good BGP attribute packing and forwarding simplicity. For unlabeled prefix, no repair label is needed. A router that is capable of handling repaired traffic is called a repair PE "rPE". The semantics of the repair label "rL" is:

- a. pop *two* labels
 - b. If "rL" is per-CE, then and send the packet to the appropriate CE
 - c. If "rL" is per-VRF, forward the packet based on the contents under the two popped labels
2. If an Egress PE knows that a P/m to which it has an external path is also reachable via another PE and that other PE advertises a repair label "rL" for P/m,
- a. It chooses the other PE as a repair PE. Let's call the chosen repair PE "rPE". The ePE chooses an IP address "rNH" local to or advertised by rPE.
 - i. "rNH" SHOULD be the next-hop attribute advertised by rPE when it announces reachability to the protected prefix P/m to minimize the number of prefixes advertised into IGP.
 - ii. if rPE also advertised a protected next-hop (pNH) for any BGP prefix that rPE can protect, then rNH MUST NOT be any protected next-hop (pNH) advertised by rPE.
 - b. Allocates a local IP address corresponding to the chosen rPE, say "pNH". "pNH" represents the protected NH. I.e. Traffic tunneled to "pNH" will be protected against edge node failure via the BGP FRR mechanism proposed in this document
 - c. A separate pNH is needed for every rPE (for a given protected PE). Each pNH must be unique within a single BGP-free core.
 - d. Now that "ePE" has a repair path for P/m, it becomes a protected PE "pPE".
 - e. Advertise pNH as a prefix into IGP
 - f. Re-advertise the protected prefix P/m to other iBGP peers with "pNH" as optional non-transitive attribute
 - g. pPE advertises the mapping (pNH,rNH) separately to all ingress PEs. A method analogous to how tunnel information is advertised [4] can be used to advertise this mapping (pNH,rNH) to ingress PE's.
 - h. Once iPE receives the pNH for each prefix and the mapping (pNH,rNH), the iPE can retrieve "rL" for P/m from the advertisement of rPE for P/m.

- i. "pPE" advertises the pair (pNH,rNH) to candidate repairing core routers.
 - j. "pPE" advertises the protected next-hop "pNH" to the penultimate hops to indicate that traffic flowing through the tunnel to the tail end "pNH" is protected against the failure of the node "pPE" and requires special processing by the penultimate hop as will be described in the next few steps
 - k. pPE advertises an explicit label for pNH instead of the usual implicit NULL. This way pPE can carry out the special label popping behavior (described in the next section if the penultimate hop cannot perform this task
3. Ingress PE "iPE"
- a. iPE receives the protected prefix P/m with "pNH" as an optional attribute
 - b. iPE also receives the mapping (pNH,rNH) from pPE
 - c. When iPE receives "rL" with P/m from rPE, then iPE can associate "rL" with P/m as described in Section 2.1.

As a result of the above steps, the following nodes store the following information

- o Ingress PE (iPE)
 - o Receives from pPE NLRI advertisement for the protected labeled prefix P/m containing the usual next-hop attribute and the optional information "pNH". iPE also receives that mapping (pNH, rNH).
 - o iPE retrieves "rL" from the advertisement of rPE for the protected prefix P/m.
 - o Assume that iPE chooses pPE as the primary NH. Then the iPE will use pNH as the tunnel tail end to pPE instead of the usual BGP next-hop
- o Penultimate Hop
 - o Receives the "pNH" from pPE
 - o As such, it knows that traffic destined to pNH needs certain special forwarding treatment as described in the next few steps

- o Penultimate hop advertises "pNH" as its own prefix but with one of the following conditions
 - . For link-state IGPs, "pNH" MAY be advertised with *maximum metric* so as not to affect the path taken by the traffic flowing from iPE's to pPE's
 - . For distance vector IGPs, the penultimate hop MAY advertise the metric of "pNH" as follows
$$\text{PHP-metric(pNH)} = \text{pPE-metric(pNH)} + \text{metric-From-PHP-to-pPE}$$
That is the metric advertised by the penultimate hop for pNH equals the metric advertised by pPE for pNH plus the metric from the penultimate hop to pPE
 - . This way the advertisement of pNH by the penultimate hop does not impact the path taken by the traffic from iPE's to pPE's
- o Repairing core router "rP" (which may also be the penultimate hop)
 - o Receives the pair (pNH,rNH) from pPE
 - o Installs the following forwarding entry for pNH
 - . If pNH is not reachable, re-tunnel traffic to rNH

2.2. Control Plane operation for Configured pNH

In Section 2.1, the pPE assigned pNH to a protected prefix P/m based on the chosen rPE. The result of this behavior is the need to re-advertise the protected prefix P/m with the associated "pNH". In this section, we outline the procedure by which the operator can pre-assign pNH to protected prefixes and hence avoid the need to re-advertise protected prefixes.

1. Protected PE "pPE"

- a. The operator groups prefixes such that two prefixes belong to the same group if the operator knows that the two prefixes are protected by the same rPE
- b. The operator assigns a distinct protected next-hop "pNH" for every group of prefixes. The assignment occurs even a repair path for P/m is not yet known.

- c. pPE advertises "pNH" as an optional non-transitive attribute with the protected prefix P/m **all the time** even if no other PE advertises P/m
- d. When pPE receives an advertisement for P/m from another PE
 - i. pPE chooses the other PE as rPE
 - ii. pPE advertises the mapping (pNH,rNH) separately to all ingress PEs. rNH SHOULD be the next-hop attribute advertised by rPE. A method analogous to how tunnel information is advertised [4] can be used to advertise this mapping (pNH,rNH) to ingress PE's.
- e. The rest of the behavior is identical to what specified in Section 2.1.

2. How to Protect the network against misconfigured pNH?

See Appendix A.

What is left is to outline the forwarding behavior before and after "pPE" failure.

2.3. Forwarding behavior at Steady State (When pPE is reachable)

This section outlines the packet forwarding procedure when pPE is still reachable

1. Ingress PE (iPE) receives a packet matching P/m and reachable via pPE
2. The iPE pushes three labels:
 - o Bottom label: VPN label advertised by pPE
 - o Second label: rL
 - o Top label: IGP label towards pNH (not the BGP next-hop attribute)
3. Penultimate Hop
 - a. Receives a packet with top label bound to pNH
 - b. Pops **two** labels **all the time**.

- c. Sends packet to pNH
- 4. Protected PE (pPE)
 - a. Receives a packet with top label as VPN label
 - b. Forwards the packet as usual
 - c. For unlabeled packets, the iPE only pushes the rL and the IGP label of pNH and the pPE uses the IP header for forwarding.

Thus the packet can be delivered correctly to its destination.

2.4. Forwarding behavior when pPE Fails

The repairing core router directly connected to a failure detects that pNH is no longer reachable. The following steps are applied.

1. Repairing core router "rP"
 - a. Receives packet with top label bound to pNH
 - b. pNH is not reachable
 - c. Swap the top label with the label of rNH
 - d. Send packet towards rPE

In effect, the repairing router re-tunnels the packet towards the repair PE
2. Penultimate hop of rPE
 - a. rNH is not a protected NH for rPE
 - b. Thus the penultimate hop employs the usual penultimate hop popping and then forwards the packet to rPE
3. Repair PE (rPE)
 - a. Receives packet with top label rL (which rPE advertised) and underneath it the regular VPN label advertised by the protected PE "pPE"
 - b. Make a lookup on "rL"
 - c. rL per CE
 - i. Pop *two* labels.

- ii. Send to correct CE
- d. rL per VRF
 - i. Pop *two* labels.
 - ii. Make IP lookup in appropriate VRF
 - iii. Send to the CE
- e. rL is assigned to unlabeled prefix
 - i. Pop "rL"
 - ii. Send the packet to the correct CE

3. Overview of the solution in a Pure IP Core

This section provides an overview of the solution when operating in a pure IP core where core routers only understand IPv4 or IPv6 protocols. Thus traffic between PEs is transported using IP tunnels such as [4][6][7].

3.1. Control Plane operation

The control plane behavior in an IP core is identical to its behavior in an MPLS core.

3.2. Forwarding Behavior at Steady State (while pPE is reachable)

1. Ingress PE (iPE) receives a packet matching P/m and reachable via pPE
2. Ingress PE:
 - o For labeled traffic, Pushes two labels
 - . Bottom label: VPN label advertised by pPE
 - . Second label: rL
 - o For unlabeled traffic, just push "rL"
 - o Encapsulates the packet into the IP tunnel header towards the pNH
3. Penultimate Hop

- o No special behavior is needed from the penultimate hop while pPE is reachable

4. Protected PE

- a. Receives an IP packet encapsulated in an IP tunnel header with destination address pNH
- b. Decapsulate the IP tunnel header and the label right under it (which will be the repair label "rL")
- c. For labeled traffic, the VPN label is exposed. So pPE makes a lookup using the VPN label. Otherwise the usual IP forwarding is applied
- d. Forwards the packet as usual

3.3. Forwarding Behavior at Failure (when pPE is not reachable)

The repairing router directly connected to a failure detects that pNH is no longer reachable. The following steps are applied.

1. Repairing router "rP"

- a. Receives IP packet with a tunnel header destined to pNH
- b. pNH is not reachable
- c. Replace the tunnel header with a tunnel header with destination address rNH
- d. Forward the packet to rNH

2. Repair PE (rPE)

- a. Receives IP packet with a tunnel header destined to rNH
- b. Decapsulate the tunnel header to expose the repair label "rL"
- c. The rest of the behavior is identical to the behavior in an MPLS Core.

4. Example

We will use an LDP core as an example. Consider the diagram depicted in Figure 2 below.

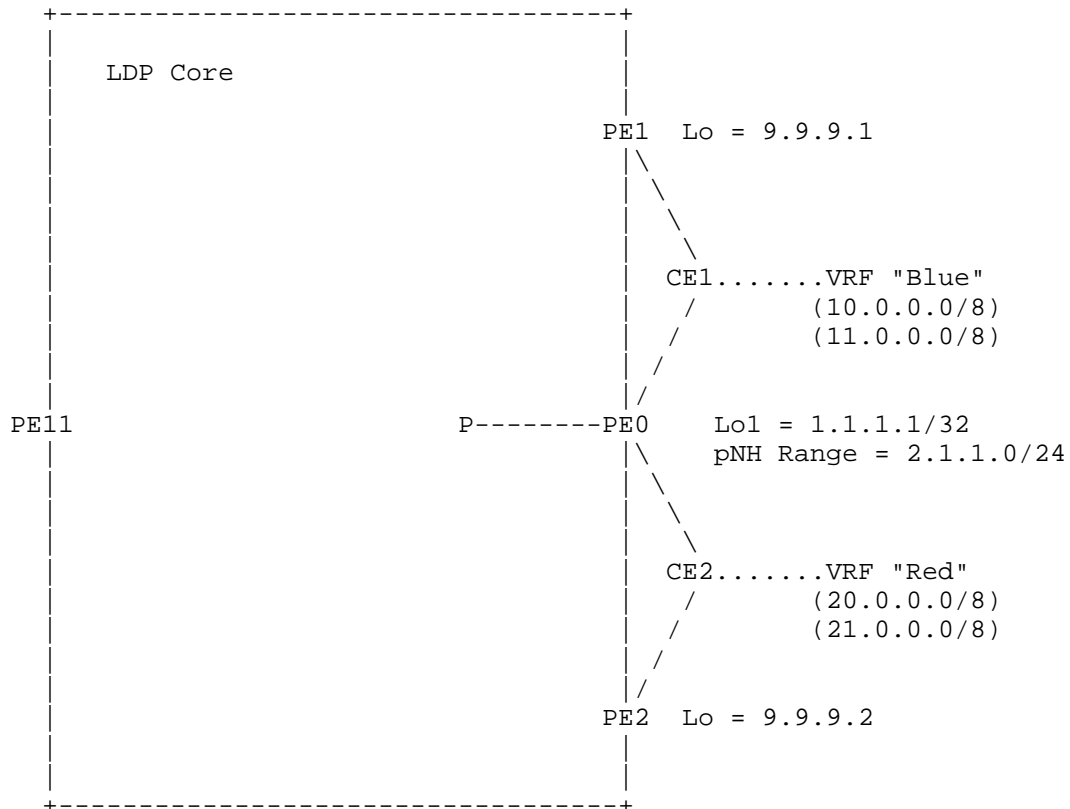


Figure 2 : Edge node BGP FRR in LDP core

- o In Figure 2, PE0 is the pPE for VRFs "Blue" and "Red" while PE1 and PE2 are the rPEs for VRFs "Blue" and "Red", respectively. VRF Blue has 10.0.0.0/8 and 11.0.0.0/8 and VRF Red has 20.0.0.0/8 and 21.0.0.0/8
- o Assuming PE0 uses per prefix label allocation, PE0 assigns the VPN labels 4100, 4200, 4300, and 4400 to 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 respectively. PE0 advertises the prefixes 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 using MP/BGP as usual

4.1. Control Plane

1. rPEs Allocate and advertise Repair labels

- a. Acting as a rPE, PE1 allocates (on per-CE basis) and advertises a repair label rL1=3100 with the prefixes 10.0.0.0/8 and 11.0.0.0/8 to all iBGP peers
- b. Similarly, PE2 allocates and advertises the repair label rL2=3200 with the prefixes 20.0.0.0/8 and 21.0.0.0/8

2. pPE calculates and advertises the pNHs

- a. For prefixes belonging to VRF "blue", PE0 allocates rNH1=2.1.1.1 because all of them are protected by PE1
- b. Similarly, for prefixes belonging to VRF "red", PE0 allocates rNH2=2.1.1.2 because VRF "red" is protected by PE2
- c. PE0 advertises (pNH1,rNH1)=(2.1.1.1, 9.9.9.1) and (pNH2,rNH2)=(2.1.1.2, 9.9.9.2) to the ingress PE PE11 and the repairing core router "P".
- d. PE0 re-advertises 10.0.0.0/8 & 11.0.0.0/8 with the optional attribute pNH1=2.1.1.1, and 20.0.0.0/8 & 21.0.0.0/8 with pNH=2.1.1.2 to the ingress PE PE11

3. The ingress PE "PE11" creates the following forwarding state

- a. For prefixes 10.0.0.0/8 & 11.0.0.0/8: Push the VPN labels 4100 and 4200, respectively, followed by rL=3100 then tunnel the packet to 2.1.1.1
- b. For prefixes 20.0.0.0/8 & 21.0.0.0/8: Push the VPN labels 4300 and 4400, respectively, followed by rL=3200; then tunnel the packet to 2.1.1.2

4.2. Forwarding Plane at Steady State (When PE0 is reachable)

1. Ingress PE PE11

- a. Traffic for VRF "Blue"
 - i. PE11 receives a packet for VRF Blue with destination address 10.1.1.1 from an external router.
 - ii. PE11 pushes the following labels
 1. The VPN label 4100

2. The Repair label 3100
3. The LDP label for the pNH 2.1.1.1
- b. Traffic for VRF "Red"
 - i. PE11 receives a packet for VRF Red with destination address 20.1.1.1 from an external router
 - ii. PE11 pushes the following labels
 1. The VPN label 4300
 2. The Repair label 3200
 3. The LDP label for the pNH 2.1.1.2
2. Penultimate Hop of PE0 (Which is also the rP "P")
 - a. Receives a packet with top label for the protected next-hop 2.1.1.1 or 2.1.1.2
 - b. Pops *2* labels
 - c. Forwards the packet to pPE which is 1.1.1.1
3. Protected PE PE0
 - a. Traffic for VRF "Blue"
 - i. PE0 receives traffic with the top label 4100.
 - ii. 4100 is the VPN label 10.1.1.1 belonging to VRF "Blue"
 - iii. PE0 pops the label 4100 and forwards the packet to CE1
 - b. Traffic for VRF "Red"
 - i. PE0 receives traffic with the top label 4300.
 - ii. 4300 is the VPN label for 20.1.1.1 belonging to VRF "Red"
 - iii. PE0 pops the label 4300 and forwards the packet to CE2
- 4.3. Forwarding Plane at Failure (When PE0 is not reachable)
 1. The ingress PE PE11

Does not know about the failure yet and hence it does not change its behavior.

2. Repair PE rP

a. Traffic for VRF "Blue"

- i. Receives a packet with the top label being the LDP label for 2.1.1.1
- ii. 2.1.1.1 is not reachable
- iii. Swap the LDP label for 2.1.1.1 with the LDP label of 9.9.9.1
- iv. Forward the packet towards 9.9.9.1

b. Traffic for VRF "Blue"

- i. Receives a packet with the top label being the LDP label for 2.1.1.2
- ii. 2.1.1.2 is not reachable
- iii. Swap the LDP label for 2.1.1.1 with the LDP label of 9.9.9.2
- iv. Forward the packet towards 9.9.9.2

3. The repair Router "PE1"

- a. The penultimate hop of PE1 performs the usual penultimate hop popping
- b. PE1 receives a packet with the top label equals the repair label 3100, which was allocated on per-CE basis and points to CE1
- c. PE1 pops *2* labels and forwards the packet to CE1

4. The repair Router "PE2"

- a. The penultimate hop of PE2 performs the usual penultimate hop popping
- b. PE1 receives a packet with the top label equals the repair label 3200, which was allocated on per-CE basis and points to CE2

c. PE2 pops *2* labels and forwards the packet to CE2

5. Inter-operability with Existing IP FRR Mechanisms

Current existing IP FRR mechanisms can be divided into two categories: core protection and edge protection. Core protection techniques, such as [12], [13], and [14], provide protection against internal node and/or link failure. Thus the technique proposed in this document is not related to existing IP FRR mechanisms. If the failure of an internal node or link results in completely disconnecting a protectable edge node, then an administrator MAY configure the repairing router to prefer the technique proposed in this document over existing IP FRR mechanisms.

Edge protection techniques, such as [16] and its practical implementation [15] provide protection against the failure of the link between PE and CE routers. Thus existing PE-CE link protection can co-exist with the techniques proposed in this document because the two techniques are independent of each other.

6. Security Considerations

No additional security risk is introduced by using the mechanisms proposed in this document

7. IANA Considerations

No requirements for IANA

8. Conclusions

This document proposes a method that allows fast re-route protection against edge node failure or complete disconnected from the core in a BGP-free core. The proposed method has few advantages

- o Easy to apply protection policies. pPE is the router that chooses the rPE. Hence if an operator wants to control what prefixes/VRFs get to be protected or what router can be chosen as repair PE, the operator needs to apply the policy on the pPE only.
- o Simple forwarding plane. The only change in forwarding plane is the need to pop/push two labels on the iPE, rP, and rPEs.

- o Single label lookup even during failure. Forwarding decisions are taken based on a single label lookup on all routers all the time even during failure
- o Immunity to mis-configuration. The only required configuration is to choose non-overlapping address ranges on different pPEs. If an operator configures overlapping IP address ranges on two different pPEs, then one of the pPE will eventually allocate a pNH that is covered by the IP address range of another pPE and hence the mis-configuration can be detected
- o No Need for IP or TE FRR: Because the exit point of the repair tunnel from rP to rPE is different from the primary tunnel exit point
- o Works in both MPLS core and IP core
- o Works with per-CE, per-VRF, and per-prefix label allocation
- o Can be incrementally deployed. There is no flag day. Different routers can be upgraded at different times
- o Zero impact on the paths taken by traffic: Enabling/deploying the feature described in this document has no effect on the paths taken by traffic at steady state

9. References

9.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006
- [3] Bates, T., Chandra, R., Katz, D., and Rekhter Y., "Multiprotocol Extensions for BGP", RFC 4760, January 2007
- [4] Malhotra, P. and Rosen, E., "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009
- [5] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.
- [6] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

- [7] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

9.2. Informative References

- [8] Marques, P., Fernando, R., Chen, E., Mohapatra, P., Gredler, H., "Advertisement of the best external route in BGP", draft-ietf-idr-best-external-04.txt, April 2011.
- [9] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.
- [10] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [11] De Clercq, J., Ooms, D., Prevost, S., Le Faucheur, F., "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, February 2007
- [12] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.
- [13] Shand, S., and Bryant, S., "IP Fast Reroute", RFC 5714, January 2010
- [14] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.
- [15] Bashandy, A., Pithawala, P., and Heitz, J., "Scalable, Loop-Free BGP FRR using Repair Label", draft-bashandy-idr-bgp-repair-label-02.txt, July 2011
- [16] O. Bonaventure, C. Filsfils, and P. Francois. "Achieving sub-50 milliseconds recovery upon bgp peering link failures," IEEE/ACM Transactions on Networking, 15(5):1123-1135, 2007

10. Acknowledgments

Special thanks to Eric Rosen, Clarence Filsfils, Maciek Konstantynowicz, Stewart Bryant, Pradosh Malhotra, Nagendra Kumar, George Swallow, Les Ginsberg, and Anton Smirnov for the valuable comments

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A.

How to protect Against Misconfigured pNH

Section 2.2 outlines a method by which the operator can configure the protected next-hop "pNH". There is a possibility of a misconfiguration as follows

- o The operator configures the same pNH for two protected prefixes P1/m1 and P2/m2 but the two prefixes are protected by different rPEs
- o The operator configures two different pNH's for two protected prefixes P1/m1 and P2/m2 but the two prefixes are protected by same rPE

The second configuration does not cause a lot of harm. Either way, routers implementing the BGP FRR scheme proposed in this document can detect both misconfigurations.

Suppose the operator configures the same "pNH" for P1/m1 and P2/m2 but P1/m1 is protected by rPE1 and P2/m2 is protected by rPE2. In that case, the iPE and misconfigured pPE will detect this inconsistency because both will see that P1/m1 and P2/m2 are assigned the same pNH but are protected by two different rPEs. The reaction to the misconfiguration is beyond the scope of this document.

Similarly, iPE and pPE can detect that the operator configured different pNH's for P1/m1 and P2/m2 even though they are protected by the same rPE because both iPE and pPE will receive an advertisement for P1/m1 and P2/m2 from the same rPE. Reactions and remedy to the misconfiguration is beyond the scope of this document.

Appendix B.
E

Alternative Approach for advertising (pNH,rNH) to iP

In Section 2.1, pPE re-advertises the protected prefixes with (pNH) as optional non-transitive attribute and advertises mapping (pNH,rNH) separately. Alternatively, iPE can re-advertise the protected prefix P/m to other iBGP peers with the mapping (pNH,rNH) as optional non-transitive attributes. Advertising (pNH) only with the prefixes has some advantages

- o Advertising pNH only with the prefixes can easily be used for configured pNH as described in Section 2.2.
- o If the repair PE changes from one PE to another, there is no need to re-advertise all the prefixes. Only the mapping (pNH,rNH) needs to be re-advertised plus possibly some of the protected prefixes
- o Advertising pNH only with the prefix slightly reduces the BGP message size

Irrespective of whether (pNH,rNH) is advertised with the prefix or separately, (pNH,rNH) is better than advertising (pNH,rL) because there are many rL's for the same rNH. Hence advertising (pNH,rNH) yields better attribute packing

Appendix C.

Modification History

C.1.1. Changes from Version 02

The whole scheme has been changed to a single next-hop per pPE-rPE. As a result, unlike version 00 and 01, there will be a need for behavioral changes in pPE, rP, iPE. The behavior for rPE remains almost unchanged

The second important change is requiring rP to advertise the pNH with maximum metric so that traffic does not get disrupted when the pPE disappears

C.1.2. Changes from Version 01

1. Use the term "underlying repair label" instead of just "repair label" to avoid confusion with the term "repair label" used in [15].
2. In version 01, it was assumed in many places that the repairing router is the penultimate hop P router. Although this would probably be the most common case, it is not always true. Hence in this version the repairing router may be any core router
3. Merged handling labeled and unlabeled prefixes into a single algorithm.
4. Allowed sending a repair label for unlabeled prefixes and added the "Push" flag. This ensures loop-free repair even for unlabeled prefixes in case that the repair PE has eiBGP paths as mentioned in Section Error! Reference source not found.
5. In Section Error! Reference source not found. discussing the rules governing the choice of the underlying repair label for labeled prefix, we changed the wording so that the primary egress PE "SHOULD" instead of "MAY" use the repair label advertised according to [15] as an underlying repair label.
6. All occurrences of the term "backup" were replaced by "repair" as the term "repair" is the commonly used term in the IP FRR context such as [14][13][12]
7. Added the definition of primary and repair tunnels in Section 1.2.
8. Added a definition of the term "Repair Next-hop" in Section 1.2.
9. Modified the definition of "repair path" in Section 1.2. to being the repair next-hop plus the underlying repair label instead of being the repair PE plus the underlying repair label.

10.Outlined inter-operability with existing IP FRR techniques in Section 5.

11.There were few editorial corrections.

Authors' Addresses

Ahmed Bashandy
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: bashandy@cisco.com

Burjiz Pithawala
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: bpithaw@cisco.com

Keyur Patel
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: keyupate@cisco.com

Network Working Group
Internet Draft
Intended status: Standards Track
Expires: April 2013

A. Bashandy
M. Konstantynowicz
N. Kumar
Cisco Systems
October 8, 2012

BGP FRR Protection against Edge Node Failure Using Table Mirroring
with Context Labels
draft-bashandy-bgp-frr-mirror-table-00.txt

Abstract

Consider a BGP free core scenario. Suppose the edge BGP speakers PE1, PE2,..., PEn know about a prefix P/m via the external routers CE1, CE2,..., CEm. If the edge router PEi crashes or becomes totally disconnected from the core, it is desirable for a core router "P" carrying traffic to the failed edge router PEi to immediately restore traffic by re-tunneling packets originally tunneled to PEi and destined to the prefix P/m to one of the other edge routers that advertised P/m, say PEj, until BGP re-converges. This draft proposes a BGP FRR scheme that relies on having the repairing edge router mirror the protected edge router forwarding table. The repairing edge router uses a locally allocated context label to identify the correct mirrored table.

Status of this Memo

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

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other

documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on April 8, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

| | |
|-----------------------------------------------------------------------------|----|
| 1. Introduction..... | 3 |
| 1.1. Conventions used in this document..... | 5 |
| 1.2. Terminology..... | 5 |
| 1.3. Problem definition..... | 7 |
| 2. Overview of BGP FRR using Mirrored Forwarding Table in an MPLS Core..... | 8 |
| 2.1. Control Plane operation..... | 8 |
| 2.2. Forwarding behavior at Steady State (while pPE is reachable) | 10 |
| 2.3. Forwarding behavior when pPE Fails..... | 10 |
| 3. Overview of the BGP FRR using Mirrored Forwarding Table in IP Core | 12 |
| 3.1. Control plane modification for IP core..... | 12 |
| 3.2. Forwarding behavior at Steady State (while pPE is reachable) | 12 |
| 3.3. Forwarding plane at Failure (when pPE is unreachable).... | 12 |
| 4. Rules for Choosing and Managing the Repair path..... | 13 |
| 5. Inter-operability with Existing IP FRR Mechanisms..... | 14 |

| | |
|----------------------------------------------------------------------------|----|
| 6. Example..... | 15 |
| 6.1. Control Plane..... | 16 |
| 6.2. Forwarding Plane at Steady State (When PE0 is reachable)..... | 17 |
| 6.3. Forwarding Plane at Failure (When PE0 is not reachable)..... | 17 |
| 7. Security Considerations..... | 19 |
| 8. IANA Considerations..... | 19 |
| 9. Conclusions..... | 19 |
| 10. References..... | 19 |
| 10.1. Normative References..... | 19 |
| 10.2. Informative References..... | 20 |
| 11. Acknowledgments..... | 21 |
| Appendix A. Auto-determination of Operating Parameters on rPE and pPE..... | 21 |
| A.1. How rPE determines the Protected PE..... | 22 |
| A.2. How pPE Determines its rPEs and Assigns pNH for each rPE..... | 22 |
| A.3. Detecting Mis-configuration..... | 23 |
| Appendix B. Ensuring correct forwarding at the edge routers..... | 24 |

1. Introduction

In a BGP free core, where traffic is tunneled between edge routers, BGP speakers advertise reachability information about prefixes to other edge routers but not to core routers. For labeled address families, namely AFI/SAFI 1/4, 2/4, 1/128, and 2/128, an edge router assigns local labels to prefixes and associates the local label with each advertised prefix such as L3VPN [11], 6PE [12], and Software [10]. Suppose that a given edge router is chosen as the best next-hop for a prefix P/m by an ingress router. The ingress router that receives a packet from an external router and destined to the prefix P/m "tunnels" the packet across the core to that egress router. If the prefix P/m is a labeled prefix, the ingress router pushes the label advertised by the egress router before tunneling the packet to the egress router. Upon receiving the packet from the core, the egress router takes the appropriate forwarding decision based on the content of the packet or the label pushed on the packet.

In modern networks, it is not uncommon to have a prefix reachable via multiple edge routers. One example is the best external path [9]. Another more common and widely deployed scenario is L3VPN [11] with multi-homed VPN sites. As an example, consider the L3VPN topology depicted in Figure 1.

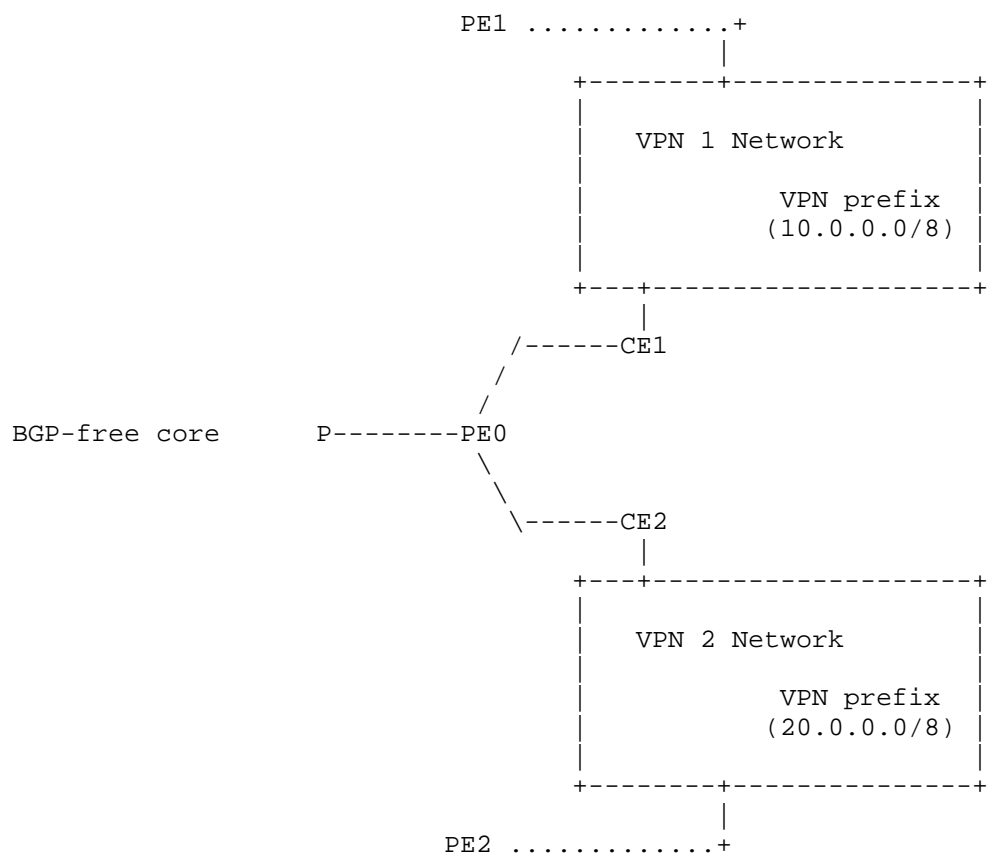


Figure 1 VPN prefix reachable via multiple PEs

As illustrated in Figure 1, the edge router PE0 is the primary NH for both 10.0.0.0/8 and 20.0.0.0/8. At the same time, both 10.0.0.0/8 and 20.0.0.0/8 are reachable through the other edge routers PE1 and PE2, respectively. On the failure of the edge router PE0, it is highly desirable for the core router P to re-route traffic for VPN 1 and VPN 2 to PE1 and PE2, respectively, without waiting for IGP or BGP to re-converge. This document proposes a scheme by which the egress and core routers participate to enable a core router to re-route traffic to the correct backup edge router when the primary edge router fails while keeping the core BGP-free

It is noteworthy to mention that the behavior specified in this draft requires supporting more than one BGP path. Methods, such as

[9], [17], and [18], may be needed to satisfy the multi-path requirement in certain scenarios such as the case were MED [2] or local preference [2] is used to determine the best path. The mechanism(s) by which a router supports BGP multi-path is beyond the scope of this document.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [1].

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

1.2. Terminology

This section defines the terms used in this document. For ease of use, we will use terms similar to those used by L3VPN [11]

- o BGP-Free core: A network where BGP prefixes are only known to the edge routers and traffic is tunneled between edge routers
- o External prefix: It is a prefix P/m (of any AFI/SAFI) that a BGP speaker has an external path for. The BGP speaker may learn about the prefix from an external peer through BGP, some other protocol, or manual configuration. The external prefix is advertised to some or all of the internal peers.
- o Protectable prefix: It is an external prefix P/m of any AFI/SAFI) that a BGP speaker has an external path to and is eligible to have a repair path.
- o Protected prefix: It is an external prefix P/m (of any AFI/SAFI) that a BGP speaker has an external path to and also has a repair path to.
- o Primary Egress PE, "ePE": It is an IBGP peer that can reach the prefix P/m through an external path and advertised the prefix to the other IBGP peers. The primary egress PE was chosen as the best path by one or more internal peers. In other words, the primary egress PE is an egress PE that will normally be used by some ingress PEs when there is no failure. Referring to Figure 1, PE0 is an egress PE.

- o Protected egress PE, "pPE" (Protected PE for simplicity): It is an egress PE for which there exists a repair path for some or all of the prefixes to which it has an external path. Referring to Figure 1, PE0 is a protected egress PE.
- o Protected edge router: Any protected egress PE.
- o Protected next-hop (pNH): It is an IPv4 or IPv6 host address belonging to the protected egress PE. Traffic tunneled to this IP address will be protected via the mechanism proposed in this document.
- o CE: It is an external router through which an egress PE can reach a prefix P/m. The routers "CE1" and "CE2" in Figure 1 are examples of such CEs.
- o Ingress PE, "iPE": It is a BGP speaker that learns about a prefix through another IBGP peer and chooses that IBGP peer as the next-hop for the prefix.
- o Repairing P router "rP" (Also "Repairing core router" and "repairing router"): A core router that attempts to restore traffic when the primary egress PE is no longer reachable without waiting for IGP or BGP to re-converge. The repairing P router restores the traffic by rerouting the traffic (through a tunnel) towards the pre-calculated repair PE when it detects that the primary egress PE is no longer reachable. Referring to Figure 1, the router "P" is the repairing P router.
- o Repair egress PE "rPE" (Repair PE for simplicity): It is an egress PE other than the primary egress PE that can reach the protected prefix P/m through an external neighbor. The repair PE is pre-calculated via other PEs prior to any failure. Referring to Figure 1, PE1 is the repair PE for 10.0.0.0/8 while PE2 is the repair PE for 20.0.0.0/8.
- o Repair next-hop (rNH): It is an IPv4 or IPv6 address belonging to the repair egress PE. If the protected prefix is advertised via BGP, then the repair next-hop SHOULD be the next-hop attribute in the BGP update message [2][3].
- o BGP nexthop (bgpNH): This is the usual next-hop attribute for route advertisements as specified in [2] and [3].
- o Context Label (cL): It is an MPLS label allocated by the repairing PE (rPE) to identify the mirrored forwarding table of the protected PE (pPE). An rPE must allocate a locally distinct context label for each mirrored forwarding table. Context labels on different rPEs may overlap

- o Repair path (Also Repair Egress Path): It is the repair next-hop.
- o Primary tunnel: It is the tunnel from the ingress PE to the primary egress PE
- o Repair tunnel: It is the tunnel from the repairing P router to the repair egress PE

1.3. Problem definition

The problem that we are trying to solve is as follows

- o Even though multiple prefixes may share the same egress router, they have different repair edge router. On losing connection to the edge router, a core router "P" detecting the loss of connection MUST reroute traffic towards the *correct* repair edge router that can reach prefixes that were reachable via the failed edge router without waiting for IGP or BGP to re-converge and update the routing tables.
- o The repairing core router P MUST NOT be forced to learn about the BGP prefixes on any of the edge router. The same applies for all core routers.
- o The size of the routing table on any core router MUST be independent of the number of BGP prefixes in the network.
- o Rerouting traffic without waiting for IGP and BGP to re-converge after a failure MUST NOT introduce loops.
- o For labeled prefixes, when a packet gets re-routed to the repair PE, the label stack on the packet MUST ensure correct forwarding.
- o At steady state, when pPE is reachable, paths taken by traffic before deploying the solution proposed in this document MUST NOT be impacted after deploying the solution proposed in this document unless desired by the operator.
- o The solution MUST be incrementally deployable
- o Minimize the number of nodes that need to be upgraded. Hence only egress PE's that participate in the solution (namely pPE's and rPE's) and protecting core routers (namely rP's) need to be upgraded.

Applying the problem to the topology in Figure 1 above, both 10.0.0.0/8 and 20.0.0.0/8 share the same primary egress router PE0,

the routing protocol(s) must identify that the node protecting repair node for 10.0.0.0/8 is PE1 while the node protecting repair node for 11.0.0.0/8 is PE2. On the failure of PE0, the core router P must reroute traffic for 10.0.0.0/8 towards PE1 and traffic for 11.0.0.0/8 towards PE2 without requiring the core router P to know about any BGP prefix.

2. Overview of BGP FRR using Mirrored Forwarding Table in an MPLS Core

The solution proposed in this document relies on the collaboration of egress PEs, and the repairing core router. This section gives an overview of how the solution works for both labeled (AFI/SAFI 1/4, 2/4, 1/128, and 2/128) and unlabeled (AFI/SAFI 1/1, 2/1, 1/2, and 2/2) protected prefixes in a core where the tunnels between edge routers are LDP LSPs [7]. Specifications of the solution in IP core are provided in Section 3.

2.1. Control Plane operation

Control plan requires certain operating parameters to be assigned. This section explains how the parameters are assigned through configuration. Automatic determination of the operating parameters is explained in Appendix A.

1. Setting the Operating parameters on pPE

- a. Suppose the protectable prefixes on a given pPE are protected by the repair edge routers rPE1, rPE2,...
- b. For the set of prefixes protected by a given rPE, assign a distinct local next-hop pNH. The pNH is also advertised as the bgpNH when the pPE advertises the prefixes to other iBGP peers. This section assumes that pNH is assigned via configuration. pNH can be automatically calculated as described in Appendix A.
- c. pNH MUST be unique within a routing domain
- d. Because pNH is also used as bgpNH, then pNH MUST be advertised into IGP as usual

2. Setting the Operating parameters on the rPE

- a. Suppose the rPE can protect prefixes whose bgpNH is pNH1, pNH2,...
- b. The operator informs rPE about the bgp next-hops that it can protect. This task can be carried out through configuration. Appendix A outlines how rPE can automatically determine the BGP next-hops it can protect.

- c. rPE performs the following tasks for each pNH
 - i. rPE allocates a "locally" distinct context label "cL" for each pNH that the rPE can protect
 - ii. rPE advertises "pNH" as its own prefix into IGP but with (maximum metric - 1) so as not to affect the path taken by the traffic flowing from iPE's to pPE's
 - iii. rPE advertises "cL" for pNH instead of implicit NULL to its neighboring LSRs. As explained in Appendix B, this behavior is necessary to ensure correct forwarding during the period starting from complete disconnect of pPE till all iPE stop using pPE as an exit point for BGP traffic.
 - iv. rPE allocates a separate "mirror" forwarding table for each pNH. The mirror forwarding table consists of a mirror IP table and a corresponding label table. The mirror table is identified by the context label "cL"
 - v. rPE assigns a local IP address rNH as the repair next-hop. rNH may be any local IP address on the rPE. "rNH" SHOULD be any next-hop attribute advertised by rPE when it announces reachability to the protected prefix P/m to minimize the number of prefixes advertised into IGP.
 - vi. rPE advertises the triplet (pNH,rNH,cL) to candidate repairing core routers. The syntax is TBD. For example, an LDP optional TLV can be used for this purpose
- d. Remember that pNH1, pNH2,... are advertised as the BGP next-hop by pPE's. When rPE receives a prefix advertisement from an iBGP peer with bgpNH equal to one of the pNHs it can protect AND rPE has at least one "external" path for the received prefix:
 - i. If the prefix is labeled ((AFI/SAFI 1/4, 2/4, 1/128, and 2/128), insert the received label into the mirror label table corresponding to the pNH
 - ii. If the prefix is unlabeled, (AFI/SAFI 1/1, 2/1, 1/2, and 2/2), insert the prefix into the mirror IP table corresponding to the pNH
 - iii. The forwarding entry of the prefix or the label in the mirror table is to either send the packet to (one of) the external path(s) or drop the packet

- iv. Remember that the external path MAY or MAY NOT be the best path. For example, if MED is used to decide the best path and the best path happened to be the internal path, then techniques, such as [9], [17], [18], and [20] are needed to calculate and advertise (an) alternative external path(s).
3. Determining the Operating Parameters on Protecting Core router "rP"
- a. rP receives the triplet (pNH,rNH,cL) from rPE
 - b. rP installs the following entry for pNH in its forwarding table
 - i. if pNH is reachable, forward the packet as usual
 - ii. If pNH is not reachable
 - 1. Swap the label bound to pNH with "cL"
 - 2. tunnel the traffic towards rNH
4. Operating parameters on the rest of the routers
- a. Other than pPE, rPE, and rP, the rest of the routers can remain totally agnostic to the BGP FRR scheme proposed in this document
 - b. Because rPE advertises pNH with (maximum-metric - 1), all the routers will prefer pPE when sending traffic to the IP address pNH. Hence as long as pPE is reachable, there is no change in traffic patterns

2.2. Forwarding behavior at Steady State (while pPE is reachable)

When pPE is reachable, there is no change in behavior due to deploying the scheme proposed in this document

2.3. Forwarding behavior when pPE Fails

The repairing router "rP" directly connected to a failure detects that pNH is no longer reachable. The following steps are applied.

- 1. Repairing router "rP"
 - a. Receives packet with top label bound to pNH
 - b. pNH is not reachable

- c. Pop the label of pNH and swap it with the context label cL received in the triplet (pNH,rNH,cL) from rPE
 - d. Push the label corresponding to rNH
 - e. Send the packet towards rNH
2. Penultimate hop of rPE performs the usual penultimate hop popping
3. Repair PE (rPE)
 - a. Because its penultimate hop performed penultimate hop popping, rPE receives a packet with the top label being the context label "cL"
 - b. rPE uses "cL" to identify the correct mirror forwarding table
 - c. rPE pops the context label "cL"
 - d. if the packet underneath "cL" is labeled, lookup the top label in the mirror label table corresponding to cL
 - e. If the packet underneath "cL" is unlabeled, lookup the destination address of the packet in the mirror IP table corresponding to cL
 - f. Forward the packet to an external neighbor or drop it based on the mirror table lookup
4. Ingress PEs (iPEs)
 - a. An ingress PE that has not yet learnt about the disappearance of pPE will continue to send traffic towards pNH and hence will be re-routed towards rPE by rP and forwarded correctly
 - b. An ingress PE that learns about the disappearance of pPE will calculate a new best path for traffic previously destined to pNH
5. The rest of the core routers
 - a. A core router that has not yet learnt that pPE is no longer reachable will continue send traffic destined to pNH towards pPE. This traffic will be intercepted by rP and re-routed towards rPE
 - b. A core router that has learnt that pPE is no longer reachable will send traffic towards rPE because rPE advertises pNH with (maximum-metric - 1).

- i. Because rPE advertises the label "cL" for rNH instead of the usual implicit NULL, a packet originally destined towards pPE that gets re-routed towards rPE will arrive at rPE with "cL" at the top
- ii. Hence rPE will process it as described in step 3.
- c. Eventually all iPEs learn that pPE is unreachable and hence no traffic will be sent to any of the pNHs advertised by pPE that has just disappeared

The next section presents the solution in an IP core.

3. Overview of the BGP FRR using Mirrored Forwarding Table in IP Core

This section describes the BGP FRR using mirrored tables solution in an IP core for both labeled (AFI/SAFI 1/4, 2/4, 1/128, and 2/128) and unlabeled (AFI/SAFI 1/1, 2/1, 1/2, and 2/2) protected prefixes.

The primary difference between a MPLS core and an IP core is that the tunnels between edge routers are IP based such as [5][6][7]. We assume that rP is capable of handling MPLS labels

3.1. Control plane modification for IP core

When using IP tunnels instead of MPLS tunnels between edge routers, there is one small modification at the repair edge router rPE. For the MPLS core, the correct mirror table at rPE is identified by the context label "cL". For the IP core, the correct mirror table must be identified by either the context label "cL" or the protected next-hop "pNH". As explained in Appendix B, this behavior is necessary to ensure correct forwarding during the period starting from complete disconnect of pPE till all iPE stop using pPE as an exit point for BGP traffic.

3.2. Forwarding behavior at Steady State (while pPE is reachable)

When pPE is reachable, there is no change in behavior due to deploying the scheme proposed in this document

3.3. Forwarding plane at Failure (when pPE is unreachable)

1. iPE is not yet aware of the failure so its behavior remains the same
2. rP

- a. Decapsulates the tunnel header towards pNH
- b. Pushes the context label "cL"
- c. Encapsulates the packet into a tunnel header with destination address rNH and forwards the packet towards rPE

3. rPE

- a. If the tunnel packet arrives with destination address "rNH"
 - i. Decapsulates the tunnel header. This exposes the context label "cL"
- b. Otherwise (i.e. the destination address is "pNH")
 - i. Decapsulate the tunnel header and associate the exposed packet with the mirror table based on "pNH"
- c. The rest of the behavior is identical to the MPLS core outlined in Section 2.3.

4. Rules for Choosing and Managing the Repair path

This section specifies rules governing how the repair path is chosen and installed in the forwarding plan. Other than the rules in this section, the method of choosing the repair path is beyond the scope of this document.

1. A repair PE MUST be another edge router that advertises the same prefix to the protected edge router pPE via IBGP peering.
2. If a repairing core router "rP" determines that the path taken by the repair tunnel to a repair edge router rPE passes through the protected edge router pPE, then the repairing router "rP" MUST NOT install this repair path in its forwarding plane. Instead, the repairing "p" router MAY use other paths that do not pass through pPE or use existing core FRR mechanisms such as [13], [14], and [15].
3. If the repair PE "rPE" advertises one or more protected next-hops, then the repair next-hop "rNH" MUST be different from any protected next-hop "pNH" advertised by rPE

If rules (1) and (2) are not applied, then the tunnel to the repair edge router rPE does not provide protection against the failure of the edge node pPE. Rule (5.) ensures that there is no ambiguity about the primary and repair next-hops

5. Inter-operability with Existing IP FRR Mechanisms

Current existing IP FRR mechanisms can be divided into two categories: core protection and edge protection. Core protection techniques, such as [13], [14], and [15], provide protection against internal node and/or link failure. Thus the technique proposed in this document is not related to existing IP FRR mechanisms. If the failure of an internal node or link results in completely disconnecting a protectable edge node, then an administrator MAY configure the repairing router to prefer the technique proposed in this document over existing IP FRR mechanisms.

Edge protection techniques, such as [16] provide protection against the failure of the link between PE and CE routers. Thus existing PE-CE link protection can co-exist with the techniques proposed in this document because the two techniques are independent of each other.

6. Example

We will use an LDP core as an example. Consider the diagram depicted in Figure 2 below.

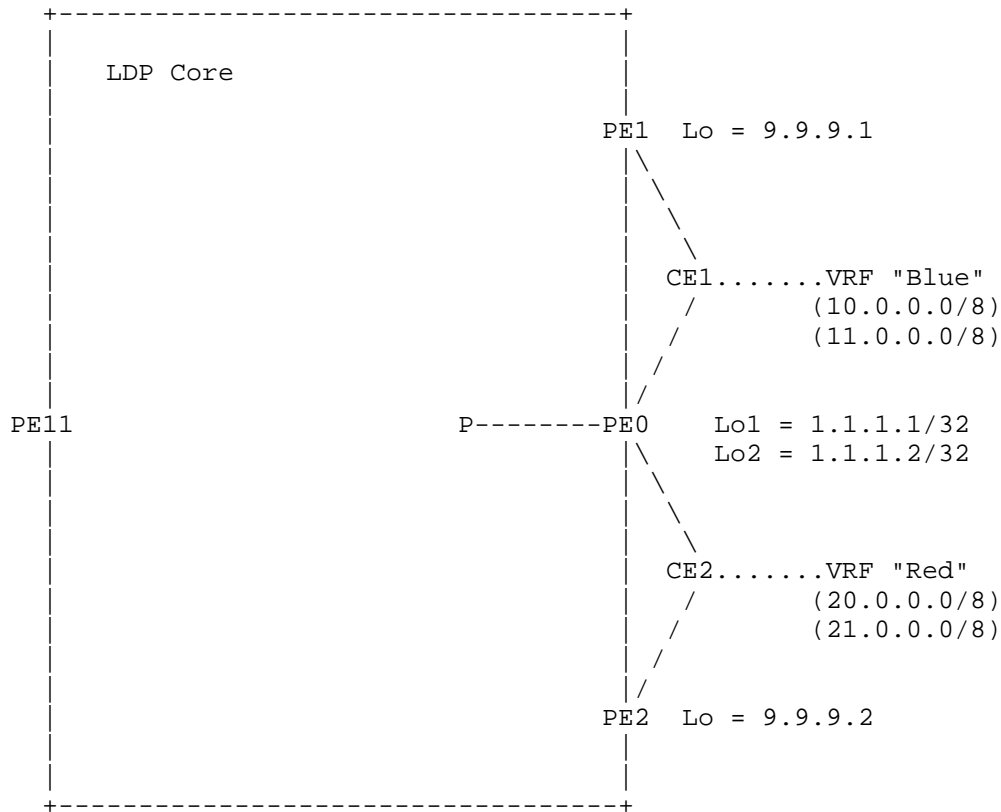


Figure 2 : Edge node BGP FRR in LDP core

- o In Figure 2, PE0 is the pPE for VRFs "Blue" and "Red". PE1 and PE2 are the rPEs for VRFs "Blue" and "Red", respectively. VRF Blue has 10.0.0.0/8 and 11.0.0.0/8 and VRF Red has 20.0.0.0/8 and 21.0.0.0/8
- o Assuming PE0 uses per prefix label allocation, PE0 assigns the VPN labels 4100, 4200, 4300, and 4400 to 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8, respectively. PE0 advertises the prefixes 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 using MP/BGP as usual

6.1. Control Plane

1. Configuring the pNHs on PE0

The operator assigns 1.1.1.1 (the IP address of Loopback0) as the bgpNH for prefixes belonging to vrf "Blue" and 1.1.1.2 (The IP address of Loopback1) as the bgpNH for prefixes belonging to vrf "Red"

2. Configuring protection parameters on rPEs

- a. The operator informs PE1 that it can protect all traffic with bgpNH=1.1.1.1. Accordingly
 - i. PE1 advertises 1.1.1.1 with (maximum-metric - 1) into IGP
 - ii. PE1 allocates a distinct mirror table for prefixes with bgpNH=1.1.1.1
 - iii. PE1 allocates the context label cL=1100 for the mirror table of bgpNH=1.1.1.1
 - iv. When advertising the FEC 1.1.1.1 to its neighboring LSRs, PE1 associates the label 1100
 - v. PE2 advertises the mapping (1.1.1.1, 9.9.9.1, 1100) to candidate repair router
 - vi. When PE1 receives a prefix advertisement from any peer with bgpNH=1.1.1.1, PE1 inserts the VPN labels in the mirror table identified by cL=1100. Hence PE1 inserts the VPN labels 4100 and 4200 in the mirror table. The forwarding entries for both labels is to either pop the label and send the packet to an external neighbor or drop the packet
- b. The operator informs PE2 that it can protect all traffic with bgpNH=1.1.1.2. Accordingly
 - i. PE2 advertises 1.1.1.2 with (maximum-metric - 1) into IGP
 - ii. PE2 allocates a distinct mirror table for prefixes with bgpNH=1.1.1.2
 - iii. PE2 allocates the context label cl=1200 for the mirror table of bgpNH=1.1.1.2
 - iv. When advertising the FEC 1.1.1.2 to its neighboring LSRs, PE2 associates the label 1200

- v. PE2 advertises the mapping (1.1.1.2, 9.9.9.2, 1200) to candidate repair router
 - vi. When PE2 receives a prefix advertisement from any peer with `bgpNH=1.1.1.2`, PE2 inserts the labels into the mirror table identified by `cL=1200`. Hence PE inserts the VPN labels 4300 and 4400 in the mirror table. The forwarding entries for both labels is to either pop the label and send the packet to an external neighbor or drop the packet
3. Enabling BGP FRR on the penultimate hop router "P"
- a. If not enabled by default, the operator enables edge node protection on the router "P"
 - b. Acting as a rP, the core router "P" receives the advertisements (`bgpNH,rNH,cL`)=(1.1.1.1, 9.9.9.1,1100) and (`bgpNH,rNH,cL`)=(1.1.1.2, 9.9.9.2,1200) from PE1 and PE2, respectively.
 - c. "rP" creates the following forwarding state for 1.1.1.1 and 1.1.1.2
 - i. If 1.1.1.1 is not reachable
 - 1. Push the context label 1100
 - 2. Send the packet through the LSP terminating on 9.9.9.1
 - ii. If 1.1.1.2 is not reachable
 - 1. Push the context label 1200
 - 2. Send the packet through the LSP terminating on 9.9.9.2

6.2. Forwarding Plane at Steady State (When PE0 is reachable)

No change in forwarding behavior when PE0 is reachable.

6.3. Forwarding Plane at Failure (When PE0 is not reachable)

- 1. Repairing core router "P"
 - a. Traffic for VRF "Blue"

- i. Receives a packet with the top label being the LDP label for 1.1.1.1
- ii. 1.1.1.1 is not reachable
- iii. Pop the LDP label of 1.1.1.1.
- iv. Push the context label 1100
- v. Push the LDP label for 9.9.9.1 and forward the packet towards PE1
- b. Traffic for VRF "Red"
 - i. Receives a packet with the top label being the LDP label for 1.1.1.2
 - ii. 1.1.1.2 is not reachable
 - iii. Pop the LDP label of 1.1.1.2.
 - iv. Push the context label 1200
 - v. Push the LDP label for 9.9.9.2 and forward the packet towards PE2
- 2. The repair Router "PE1"
 - a. The penultimate hop of PE1 performs the usual penultimate hop popping
 - b. PE1 receives a packet with the top label equals the context label 1100
 - c. PE1 makes a lookup for 1100 in its label table. The lookup yields the mirror table of the bgpNH=1.1.1.1
 - d. Pop the cL=1100. This exposes the VPN label 4100 or 4200.
 - e. Lookup VPN label 4100 or 4200 in the mirror table corresponding to cL=1100. The lookup results in popping the VPN label 4100 or 4200 and forwarding the packet natively to CE2
- 3. The repair Router "PE2"
 - a. The penultimate hop of PE2 performs the usual penultimate hop popping

- b. PE2 receives a packet with the top label equals the context label 1200
- c. PE2 makes a lookup for 1200 in its label table. The lookup yields the mirror table of the bgpNH=1.1.1.2
- d. Pop the cL=1200. This exposes the VPN label 4300 or 4400
- e. Lookup the VPN label 4300 or 4400 in the mirror table. The lookup results in popping the VPN label 4300 or 4400 and forwarding the packet natively to CE2

7. Security Considerations

No additional security risk is introduced by using the mechanisms proposed in this document

8. IANA Considerations

No requirements for IANA

9. Conclusions

This document proposes a method that allows fast re-route protection against edge node failure or complete disconnected from the core in a BGP-free core. The method does not require support of LFA FRR [13][14][15] and most of the provisioning effort can be automated at the expense of the possible need to re-advertise prefixes as described in Appendix A.

10. References

10.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006
- [3] Bates, T., Chandra, R., Katz, D., and Rekhter Y., "Multiprotocol Extensions for BGP", RFC 4760, January 2007
- [4] Malhotra, P. and Rosen, E., "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009

- [5] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.
- [6] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.
- [7] L. Andersson, I. Minei, B. Thomas, "LDP Specifications", RFC 5036, October 2007
- [8] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

10.2. Informative References

- [9] Marques, P., Fernando, R., Chen, E., Mohapatra, P., Gredler, H., "Advertisement of the best external route in BGP", draft-ietf-idr-best-external-04.txt, April 2011.
- [10] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.
- [11] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [12] De Clercq, J., Ooms, D., Prevost, S., Le Faucheur, F., "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, February 2007
- [13] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.
- [14] Shand, S., and Bryant, S., "IP Fast Reroute", RFC 5714, January 2010
- [15] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.
- [16] O. Bonaventure, C. Filsfils, and P. Francois. "Achieving sub-50 milliseconds recovery upon bgp peering link failures, " IEEE/ACM Transactions on Networking, 15(5):1123-1135, 2007
- [17] D. Walton, E. Chen, A. Retana, J. Scudder, "Advertisement of Multiple Paths in BGP", draft-ietf-idr-add-paths-07.txt, June 2012
- [18] R. Raszuk, R. Fernando, K. Patel, D. McPherson, K. Kumaki, "Distribution of diverse BGP paths", draft-ietf-grow-diverse-bgp-path-dist-08.txt, July 2012

- [19] T. Bates, E. Chen, and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC4456, Apr 2006
- [20] P. Mohapatra, R. Fernando, C. Filsfils, and R. Raszuk, "Fast Connectivity Restoration Using BGP Add-path", draft-pmohapat-idr-fast-conn-restore-02, October 2011

11. Acknowledgments

Special thanks to Clarence Filsfils, Eric Rosen, Stewart Bryant, and Pradosh Mohapatra for the valuable comments

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A. Auto-determination of Operating Parameters on rPE and pPE

The main provisioning effort as outlined in Section 2 is the assignment of a domain-wide distinct pNH for each rPE-pPE pair and configuring the pNH on the correct pPE and rPE. This section outlines a method by which the assignment of pNH to rPE on a given pPE is automated thereby eliminating the need for any operator intervention except for configuring the range of IP addresses from which pNHs are taken. The automation comes at the expense of the need to re-advertise BGP prefixes under certain conditions as outlined below in this Section.

The objective of the automation is to

- o Let the rPE determine which pPEs the rPE can protect and hence assign a local context label "cL" for each pPE and mirror the portion of the pPE routing table that rPE can protect (Remember that rPE can protect a prefix advertised by pPE if rPE has an external path for that prefix)
- o Let the pPE determine which PEs can act as rPE's for some or all of its prefixes and hence automatically assign a pNH for each distinct rPE out of a preconfigured range of IP addresses

When PEs peer directly with each other, it is easy to determine the router ID of the advertising router. In the presence of a router reflector [19], it is not possible to directly determine the router ID of the advertising PE. Hence we introduce the "RID-attr" optional non-transitive attribute. The actual format of the "RID-attr" attribute is TBD. It contains the router ID of the advertising PE.

Each PE MUST have a distinct router ID within a routing domain. "RID-attr" MUST be advertised with each protectable prefix.

A.1. How rPE determines the Protected PE

Assuming that the "RID-attr" is advertised as an optional attribute with all protectable prefixes, the rPE applies the following steps to determine the pPE

1. rPE receives route advertisements from another peer and the advertisement includes the peer's RID in the optional attribute "RID-Attr"
2. If rPE has an external path for some or all of the received route advertisements and rPE advertises some or all these route advertisements (as best paths or otherwise such as [9], [17], and [18]), then it considers the peer as a pPE
 - a. rPE allocates a distinct context "cL" label for the pPE
 - b. rPE advertises the mapping cL-->RID all the time to all peers. The syntax is TBD for the time being but a method similar to advertising tunnel information [4] can be used
3. If rPE loses all external paths for all prefixes from the peer identified by "RID", then rPE withdraws the mapping "cL-->RID"
4. If rPE cannot protect all routes advertised by the pPE but can protect some of them, then rPE re-advertises the protectable prefixes it previously advertise but attaches the context label "cL" as a non-transitive optional attribute. The syntax of "cL" is TBD. This is one of the cases where prefixes previously advertised need to be re-advertised
5. rPE creates a mirror table for pPE. If rPE can protect a route received from pPE, then rPE mirrors that route into the mirror table for pPE

A.2. How pPE Determines its rPEs and Assigns pNH for each rPE

1. When pPE receives the mapping cL-->RID where RID is the router ID of the pPE, pPE assumes the router that advertised the mapping cL-->RID is an rPE
2. pPE allocates a distinct pNH for the rPE

3. The next step is for pPE to re-advertise some or all of its prefixes but use the pNH assigned to rPE as bgpNH. Let $\{P1/m1, \dots, Pk/mk\}$ be the set prefix that rPE advertises to its peers (as best paths or otherwise such as [9], [17], and [18]) and, at the same time, pPE advertises as reachable prefixes in the the NLRI field. There are two cases
 - a. Case 1: rPE advertises the mapping cL-->RID but rPE does not associate the context label "cL" as an optional attribute with any prefix $\{P1/m1, \dots, Pk/mk\}$
 - b. Case 2: rPE advertises the mapping cL-->RID and rPE associates "cL" as an optional attribute with a *subset* of the prefixes $\{P1/m1, \dots, Pk/mk\}$
4. Case 1: rPE does not associate the context label "cL" with advertised prefixes. In that case, pPE assumes that rPE can protect all of the prefixes $\{P1/m1, \dots, Pk/mk\}$. Hence pPE re-advertises $\{P1/m1, \dots, Pk/mk\}$ uses the pNH assigned for the rPE as bgpNH.
5. Case 2: rPE associates "cL" with a *subset* of $\{P1/m1, \dots, Pk/mk\}$. In that case, pPE assumes the rPE can only protect the subset of $\{P1/m1, \dots, Pk/mk\}$ that has "cL". Hence rPE re-advertises this subset but uses the pNH assigned for the rPE as bgpNH.
6. Cases 1 and 2 are the second case where prefixes previously advertised are re-advertised without any topology changes

A.3. Detecting Mis-configuration

The auto assignment of pNH described in this appendix still requires the operator to configure a range of IP addresses from which a pPE allocates the protected next-hops "pNH". Because the pNH allocated by two different pPEs MUST NOT be identical, then the range of IP addresses on two different pPEs MUST NOT overlap. Hence the only possible misconfiguration is configuring overlapping IP ranges on two different pPE. This section describes how such misconfiguration can be detected. Suppose pPE1 and pPE2 where configured with overlapping IP ranges. Such misconfiguration can be detected as follows:

1. Because in case of misconfiguration the IP ranges on pPE1 and pPE2 overlap, then at one point in time, pPE1 will allocate a pNH that falls within the IP range configured on pPE2
2. As described in Section A.2 pPE1 re-advertises some or all of its prefixes and use the allocated pNH as the bgpNH attribute

3. When pPE2 receives an advertisement from another peer containing a bgpNH within pPE2's configured IP range, then pPE2 detects the misconfiguration

Appendix B. Ensuring correct forwarding at the edge routers

As mentioned in Section 2 both rPE and pPE advertise the protected next-hop "pNH" in the core. To ensure no impact on traffic engineering, rPE advertises "pNH" with (max-metric - 1). When the primary edge router pPE becomes totally disconnected from the core, some core routers may start to forward traffic originally destined to pPE to rPE. Thus it is possible that traffic originally destined to pPE arrives at rPE without "cL" appearing at the top of the label stack. The behavior explained in Section 2 for MPLS core and Section 3 for IP core ensures that traffic is forwarded correctly when arriving at rPE.

In an MPLS core, the rPE advertises the label "cL" for pNH. Hence traffic originally destined for pNH and re-routed by a core router towards rPE will arrive at rPE with "cL" at the top. Hence rPE can identify the correct mirror table and be able forward the packet correctly

In an IP core, rPE associates the IP address "pNH" with the mirror table. Hence if a core router re-routes traffic originally tunneled towards pPE to rPE, the tunnel packets arrive at rPE with the destination address "pNH". This allows rPE to identify the correct mirror table and be able to forward the packet correctly

Authors' Addresses

Ahmed Bashandy
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: bashandy@cisco.com

Maciek Konstantynowicz
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: mkonstan@cisco.com

Nagendra Kumar
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: naikumar@cisco.com

Network Working Group
Internet Draft
Intended status: Standards Track
Expires: January 2013

A. Bashandy
N. Kumar
M. Konstantynowicz
Cisco Systems
July 7, 2012

BGP FRR Protection against Edge Node Failure Using Vector Labels
draft-bashandy-bgp-frr-vector-label-00.txt

Abstract

Consider a BGP free core scenario. Suppose the edge BGP speakers PE1, PE2,..., PEn know about a prefix P/m via the external routers CE1, CE2,..., CEm. If the edge router PEi crashes or becomes totally disconnected from the core, it is desirable for a core router "P" carrying traffic to the failed edge router PEi to immediately restore traffic by re-tunneling packets originally tunneled to PEi and destined to the prefix P/m to one of the other edge routers that advertised P/m, say PEj, until BGP re-converges. In doing so, it is highly desirable to keep the core BGP-free while not imposing restrictions on external connectivity or complicating provisioning effort. Thus (1) a core router should not be required to learn any BGP prefix, (2) the size of the forwarding and routing tables in the core routers should be independent of the number of BGP prefixes, (3) re-routing traffic without waiting for re-convergence must not cause loops, (4) provisioning effort should be kept at minimum, and (5) there should be no restrictions on what edge routers advertise what prefixes. For labeled prefixes, (6) the label stack on the packet must allow the repair PEj to correctly forward the packet and (7) there must not be any need to perform more than one label lookup on any edge or core router during steady state

Status of this Memo

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

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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

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

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

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

This Internet-Draft will expire on January 7, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

| | |
|----------------------------------------------------------------------------|----|
| 1. Introduction..... | 3 |
| 1.1. Conventions used in this document..... | 5 |
| 1.2. Terminology..... | 5 |
| 1.3. Problem definition..... | 7 |
| 2. Overview of BGP FRR in an MPLS Core..... | 8 |
| 2.1. Control Plane operation..... | 8 |
| 2.2. Forwarding behavior at Steady State (while pPE is reachable) | 12 |
| 2.3. Forwarding behavior when pPE Fails..... | 13 |
| 3. Overview of the BGP FRR using Vector Labels in an IP Core..... | 14 |
| 3.1. Pure IP Core..... | 15 |

| | |
|-------------------------------------------------------------------------------------|----|
| 3.1.1. Control Plane..... | 15 |
| 3.1.2. Forwarding plane during Steady State (when pPE is reachable)..... | 15 |
| 3.1.3. Forwarding plane at Failure (when pPE is unreachable) | 15 |
| 3.2. Hybrid IP core..... | 16 |
| 3.2.1. Control Plane..... | 16 |
| 3.2.2. Forwarding Plane during Steady State (when pPE is reachable)..... | 17 |
| 3.2.3. Forwarding plane at Failure (when pPE is unreachable) | 17 |
| 4. Rules for Choosing and Managing the Repair path..... | 17 |
| 4.1. General Rules for Managing the Repair Path..... | 18 |
| 4.2. Rules for Choosing the Repair Path for Labeled Prefixes.. | 19 |
| 5. Inter-operability with Existing IP FRR Mechanisms..... | 19 |
| 6. Example..... | 20 |
| 6.1. Control Plane..... | 21 |
| 6.2. Forwarding Plane at Steady State (When PE0 is reachable). | 22 |
| 6.3. Forwarding Plane at Failure (When PE0 is not reachable).. | 24 |
| 7. Security Considerations..... | 25 |
| 8. IANA Considerations..... | 25 |
| 9. Conclusions..... | 25 |
| 10. References..... | 26 |
| 10.1. Normative References..... | 26 |
| 10.2. Informative References..... | 27 |
| 11. Acknowledgments..... | 27 |
| Appendix A. Other Algorithms to Allocate and Disseminate Vector labels..... | 28 |
| A.1. iPE chooses the repair path..... | 28 |
| A.1.1. Allocating Vector Labels using a Hash Function..... | 28 |
| A.1.1.1. Calculating and distributing the mapping rNH->vL to different routers..... | 28 |
| A.1.1.1.2. Risk of Mis-configuration leading to Mismatch in rNH-->vL Mapping..... | 29 |
| A.1.1.1.3. Risk of forwarding to Incorrect VRF during convergence only..... | 29 |
| A.1.2. pPE Allocates and advertises vL with protected prefixes | 29 |
| A.1.2.1.1. Risk of forward to Incorrect VRF during Convergence Only..... | 30 |
| A.2. pPE chooses rPE and distributes the mapping of vL-->rNH.. | 30 |
| A.3. Combination of iPE and pPE Choosing rPE..... | 31 |

1. Introduction

In a BGP free core, where traffic is tunneled between edge routers, BGP speakers advertise reachability information about prefixes to other edge routers but not to core routers. For labeled address families, namely AFI/SAFI 1/4, 2/4, 1/128, and 2/128, an edge router assigns local labels to prefixes and associates the local label with each advertised prefix such as L3VPN [10], 6PE [11], and

Software [9]. Suppose that a given edge router is chosen as the best next-hop for a prefix P/m. An ingress router that receives a packet from an external router and destined to the prefix P/m "tunnels" the packet across the core to that egress router. If the prefix P/m is a labeled prefix, the ingress router pushes the label advertised by the egress router before tunneling the packet to the egress router. Upon receiving the packet from the core, the egress router takes the appropriate forwarding decision based on the content of the packet or the label pushed on the packet.

In modern networks, it is not uncommon to have a prefix reachable via multiple edge routers. One example is the best external path [8]. Another more common and widely deployed scenario is L3VPN [10] with multi-homed VPN sites. As an example, consider the L3VPN topology depicted in Figure 1.

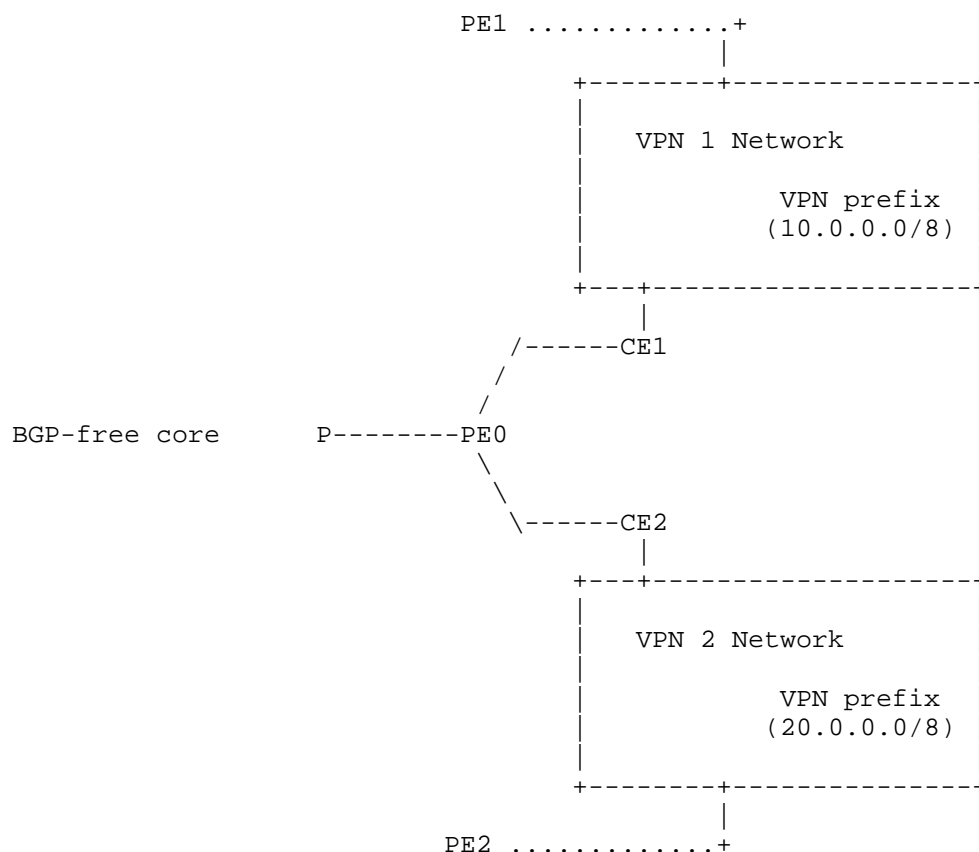


Figure 1 VPN prefix reachable via multiple PEs

As illustrated in Figure 1, the edge router PE0 is the primary NH for both 10.0.0.0/8 and 20.0.0.0/8. At the same time, both 10.0.0.0/8 and 20.0.0.0/8 are reachable through the other edge routers PE1 and PE2, respectively.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [1].

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

1.2. Terminology

This section defines the terms used in this document. For ease of use, we will use terms similar to those used by L3VPN [10]

- o BGP-Free core: A network where BGP prefixes are only known to the edge routers and traffic is tunneled between edge routers
- o External prefix: It is a prefix P/m (of any AFI/SAFI) that a BGP speaker has an external path. The BGP speaker may learn about the prefix from an external peer through BGP, some other protocol, or manual configuration. The protected prefix is advertised to some or all of the internal peers.
- o Protectable prefix: It is an external prefix P/m of any AFI/SAFI) that a BGP speaker has an external path to and is eligible to have a repair path.
- o Protected prefix: It is an external prefix P/m (of any AFI/SAFI) that a BGP speaker has an external path to and also has a repair path to.
- o Primary Egress PE, "ePE": It is an IBGP peer that can reach the prefix P/m through an external path and advertised the prefix to the other IBGP peers. The primary egress PE was chosen as the best path by one or more internal peers. In other words, the primary egress PE is an egress PE that will normally be used by some ingress PEs when there is no failure. Referring to Figure 1, PE0 is an egress PE.

- o Protected egress PE, "pPE" (Protected PE for simplicity): It is an egress PE for which there exists a repair path for some or all of the prefixes to which it has an external path. Referring to Figure 1, PE0 is a protected egress PE.
- o Protected edge router: Any protected egress PE.
- o Protected next-hop (pNH): It is an IPv4 or IPv6 host address belonging to the protected egress PE. Traffic tunneled to this IP address will be protected via the mechanism proposed in this document. Note that, in most cases, the protected next-hop will be different from the next-hop attribute in the BGP update message [2][3].
- o CE: It is an external router through which an egress PE can reach a prefix P/m. The routers "CE1" and "CE2" in Figure 1 are examples of such CEs.
- o Ingress PE, "iPE": It is a BGP speaker that learns about a prefix through another IBGP peer and chooses that IBGP peer as the next-hop for the prefix.
- o Repairing P router "rP" (Also "Repairing core router" and "repairing router"): A core router that attempts to restore traffic when the primary egress PE is no longer reachable without waiting for IGP or BGP to re-converge. The repairing P router restores the traffic by rerouting the traffic (through a tunnel) towards the pre-calculated repair PE when it detects that the primary egress PE is no longer reachable. Referring to Figure 1, the router "P" is the repairing P router.
- o Repair egress PE "rPE" (Repair PE for simplicity): It is an egress PE other than the primary egress PE that can reach the protected prefix P/m through an external neighbor. The repair PE is pre-calculated via other PEs prior to any failure. Referring to Figure 1, PE1 is the repair PE for 10.0.0.0/8 while PE2 is the repair PE for 20.0.0.0/8.
- o Underlying Repair label (rL): The underlying repair label is the label that is advertised by rPE and is used by rPE to forward repaired traffic, which is traffic re-tunneled by the rP after detecting that the pPE is no longer reachable. A repair label is defined for labeled protected prefixes only.
- o Repair next-hop (rNH): It is an IPv4 or IPv6 host address belonging to the repair egress PE. If the protected prefix is advertised via BGP, then the repair next-hop MAY be the next-hop attribute in the BGP update message [2][3].

- o BGP nexthop (bgpNH): This is the usual next-hop attribute for route advertisements as specified in [2]in [3]. In most case, bgpNH is different from pNH
- o Vector Label (vL): It is a label that identifies the repair PE within a certain label context. Every distinct rPE must have a distinct vector label the aforementioned label context. Vector labels in different label contexts may overlap
- o Repair path (Also Repair Egress Path): It is the repair next-hop. If an underlying repair label exists, the repair path is the repair next-hop together with the underlying repair label.
- o Primary tunnel: It is the tunnel from the ingress PE to the primary egress PE
- o Repair tunnel: It is the tunnel from the repairing P router to the repair egress PE

1.3. Problem definition

The problem that we are trying to solve is as follows

- o Even though multiple prefixes may share the same egress router, they have different repair edge router. In Figure 1 above, both 10.0.0.0/8 and 20.0.0.0/8 share the same primary next hop PE0, the routing protocol(s) must identify that the node protecting repair node for 10.0.0.0/8 is PE1 while the node protecting repair node for 11.0.0.0/8 is PE2
- o On loosing connection to the edge router, the core router "P" MUST reroute traffic towards the *correct* repair edge router that can reach prefixes that were reachable via the failed edge router without waiting for IGP or BGP to re-converge and update the routing tables. On the failure of PE0 illustrated in Figure 1, the core router P needs to reroute traffic for 10.0.0.0/8 towards PE1 and traffic for 11.0.0.0/8 towards PE2
- o The repairing core router P MUST NOT be forced to learn about the BGP prefixes on any of the edge router. The same applies for all core routers.
- o There SHOULD NOT be a need for a special router or group of routers to handle rerouting traffic on edge node failure.
- o The size of the routing table on any core router MUST be independent of the number of BGP prefixes in the network.

- o Rerouting traffic without waiting for IGP and BGP to re-converge after a failure MUST NOT cause loops.
- o For labeled prefixes, when a packet gets re-routed to the repair PE, the label stack on the packet MUST ensure correct forwarding.
- o Provisioning and maintenance overhead must be kept at minimum
- o At steady state, when pPE is reachable, paths taken by traffic must not be impacted by deploying the solution proposed in this document unless desired by the operator.
- o The solution must be incrementally deployable

2. Overview of BGP FRR in an MPLS Core

The solution proposed in this document relies on the collaboration of egress PE, ingress PE, penultimate hop routers, and repairing core router. This section gives an overview of how the solution works for both labeled (AFI/SAFI 1/4, 2/4, 1/128, and 2/128) and unlabeled (AFI/SAFI 1/1, 2/1, 1/2, and 2/2) protected prefixes in an MPLS core. Specifications of the solution in IP core are provided in Section 3.

2.1. Control Plane operation

1. Each egress router that is capable of handling repaired traffic assigns each protectable labeled prefix a repair label: "rL". "rL" is advertised as optional path attribute. "rL" MUST be Per-CE or per-VRF for good BGP attribute packing and forwarding simplicity. For unlabeled prefix, no repair label is needed. A router that is capable of handling repaired traffic is called a repair PE "rPE".
 - a. The semantics of the repair label "rL" is:
 - i. If "rL" is per-CE, then pop *two* labels and send the packet to the appropriate CE
 - ii. If "rL" is per-VRF, then pop *two* labels and forward the packet based on the contents under the two popped labels
2. Each protectable egress PE (pPE) is assigned a unique protectable IP address "pNH". Traffic tunneled to pNH is protected by the BGP FRR proposed in this document
 - a. Only a single pNH is needed per pPE

- b. If all iPE's support the BGP FRR scheme proposed in this document, then pNH may be the usual BGP next-hop attribute. Otherwise, pNH MUST NOT be identical to the BGP next-hop attribute
 - c. pPE advertises pNH as a prefix into IGP
 - d. pPE advertises an explicit label for pNH (instead of the usual implicit NULL). This way if the penultimate hop does not understand the BGP FRR scheme proposed in this document, pPE can handle the special popping behavior for protected traffic tunneled to pNH
 - e. "pPE" advertises the protected next-hop "pNH" to the penultimate hops to indicate that traffic flowing through the tunnel to the tail end "pNH" is protected against the failure of the node "pPE" and requires special processing by the penultimate hop as will be described in the next few steps
 - f. For every BGP next-hop (bgpNH) that pPE advertises with its routes, pPE separately advertises the mapping (bgpNH,pNH) to all ingress PE. A method analogous to how tunnel information is advertised [4] can be used to advertise this mapping to ingress PE's. The mapping "(bgpNH,pNH)" means: if the ingress PE wants to protect traffic normally tunneled to "bgpNH" against the failure of "pPE", the iPE MUST tunnel the traffic to "pNH" instead of bgpNH.
3. If a pPE knows that a P/m to which it has an external path is also reachable via another PE,
- a. pPE chooses one of the other PEs as a repair PE "rPE". The pPE chooses, as a repair next-hop, an IP address "rNH" local to or advertised by rPE. Rules governing rNH are
 - i. "rNH" SHOULD be the next-hop attribute advertised by rPE when it announces reachability to the protected prefix P/m to minimize the number of prefixes advertised into IGP and BGP.
 - ii. if rPE also advertised a protected next-hop (pNH) for any BGP prefix that rPE can protect, then rNH MUST NOT be any protected next-hop (pNH) advertised by rP
 - b. pPE assigns a vector label "vL" for "rNH". A distinct "vL" is needed for every distinct "rNH" within the context of a pPE

- c. pPE advertises the mapping (pNH,rNH,vL) to all ingress PE's. The mapping (pNH,rNH,vL) means: "Within the context of the protected next-hop pNH, the repair next-hop rNH is assigned the vector label vL"
 - d. "pPE" advertises the triplet (pNH,rNH,vL) to candidate repairing core routers. For example, an LDP optional TLV can be used for this purpose
4. An ingress PE "iPE" receives route updates from pPE with "bgpNH" as the next-hop attribute. Suppose an ingress PE "iPE" chooses "bgpNH" as the best path for one or more protectable PE. If iPE wants to protect traffic tunneled to "bgpNH" against pPE failure, "iPE" performs the following steps
- a. iPE receives the mapping (bgpNH,pNH) from pPE to indicate that the protected next-hop for traffic tunneled to bgpNH is pNH
 - b. iPE receives the mapping (pNH,rNH,vL) from "pPE" to indicate that the vector label pointing to the repair next-hop "rNH" for traffic tunneled to pNH is "vL"
 - c. iPE receives an advertisement for the protectable route from rPE with "rNH" as the next-hop
 - d. If the above 3 conditions are satisfied, then iPE chooses rPE as the repair PE with rNH as the repair next-hop and the vector label "vL"

As a result of the above steps, the following nodes store the following information

- o Ingress PE (iPE)
 - o Receives from pPE NLRI advertisement for the protected labeled prefix P/m containing the usual BGP next-hop attribute "bgpNH"
 - o Receives from pPE the mapping (bgpNH,pNH). This means that if iPE wants to protect traffic normally tunneled to "bgpNH" against pPE failure, the iPE MUST tunnel the traffic to "pNH" instead of "bgpNH"

- o Receives the triplet (pNH,rNH,vL). The triplet (pNH,rNH,vL) means that if iPE chooses rNH as the repair next-hop for the traffic tunneled to the protected next-hop pNH, then iPE has to use the vector label "vL" while tunneling traffic. The method of using the vector label "vL" is described in the forwarding behavior in Section 2.2 and 2.3.
- o Penultimate Hop
 - o Receives the "pNH" from pPE
 - o As such, it knows the pNH needs certain special treatment as described in the forwarding behavior in Section 2.2 and 2.3.
 - o Penultimate hop advertises "pNH" as its own prefix into IGP. The penultimate hop advertises pNH so that when pPE is lost, nodes continue to forward the traffic towards the original pPE and hence get protected by the rP. This behavior is required until BGP on the iPE's recalculate and start forwarding traffic towards an alternative PE.
 - o Penultimate hop advertises "pNH" as its own prefix into IGP but with one of the following conditions
 - . For link-state IGPs, "pNH" MAY be advertised with *maximum metric* so as not to affect the path taken by the traffic flowing from iPE's to pPE's
 - . For distance vector IGPs, the penultimate hop advertises metric of "pNH" as follows
$$\text{PHP-metric(pNH)} = \text{pPE-metric(pNH)} + \text{metric-From-PHP-to-pPE}$$
That is, the metric advertised by the penultimate hop for pNH equals the metric advertised by pPE for pNH plus the metric from the penultimate hop to pPE
- . This way the advertisement of pNH by the penultimate hop into IGP does not impact the path taken by the traffic from iPE's to pPE's

- . When does the penultimate hop stop advertising pNH as its own prefix? The penultimate hop should continue to advertise pNH long enough for iPE's to re-converge. Advertising pNH longer than necessary is harmless because iPE's would have already re-converged to a new BGP next-hop and hence no traffic will be attracted to the non-existing pNH. The specific period length can be subject to configuration but the default value may be in the order of 2-3 minutes
- o Repairing core router "rP" (which may also be the penultimate hop)
 - o Receives the triplet (pNH,rNH,vL) from pPE
 - o Creates a distinct label context for "pNH"
 - . In LDP core, the context is identified by the IGP label of pNH
 - . In an IP core, the context is identified by the "pNH" address itself.
 - o Inserts the label vL in the label context identified by pNH.
 - . The forwarding entry for vL in the label context of pNH is
 - . Swap vL with the IGP label of rNH
 - . Forward the packet towards rNH
 - o Installs the following forwarding entry for pNH
 - . If pNH is not reachable, pop the label for pNH and lookup the label underneath the label of pNH in the label context of pNH
 - . Otherwise, forward the packet to pNH as usual

What is left is to outline the forwarding behavior before and after the failure of "pNH".

2.2. Forwarding behavior at Steady State (while pPE is reachable)

This section outlines the packet forwarding procedure when pPE is still reachable.

1. Ingress PE (iPE) receives a packet matching P/m from an external neighbor and reachable via pPE

2. Ingress PE: Pushes **four** labels

- o Bottom label: VPN label advertised by pPE
- o Second label: rL
- o Third label: vL corresponding to chosen rNH
- o Top label: IGP label towards pNH (not the bgpNH attribute)
- o In pushing the labels "vL" following by "rL", iPE practically encodes the chosen repair path into the packet.

3. Penultimate Hop

- a. Receives a packet with top label bound to pNH
- b. Pops **three** labels **all the time**.
- c. Sends packet to pNH

4. Protected Egress PE (pPE)

- a. Receives a packet with top label as VPN label
- b. Forwards the packet as usual

Thus the packet can be delivered correctly to its destination.

2.3. Forwarding behavior when pPE Fails

The repairing router "rP" directly connected to a failure detects that pNH is no longer reachable. The following steps are applied.

1. Repairing router "rP"

- a. Receives packet with top label bound to pNH
- b. pNH is not reachable
- c. Pop the label of pNH. The vector label "vL" is right under the label of pNH
- d. Lookup "vL" in the label context identified by the label of "pNH". The lookup yields a rewrite label corresponding to the chosen rNH
- e. Swap the top label with the label of rNH

- f. Send packet towards rNH
 - g. In effect, the repairing router uses the vector label to find the repair PE chosen by the ingress PE
2. Penultimate hop of rPE
- a. rNH is not a protected NH for rPE
 - b. Thus the penultimate hop employs the usual penultimate (single label) hop popping and then forwards the packet to rPE
3. Repair PE (rPE)
- a. Receives packet with top label rL (which rPE advertised) and the bottom label is the regular VPN label advertised by the primary PE "pPE"
 - b. Make a lookup on "rL"
 - c. rL per CE
 - i. Pop *two* labels.
 - ii. Send to correct CE
 - d. rL per VRF
 - i. Pop *two* labels.
 - ii. Make IP lookup in appropriate VRF
 - iii. Send to the CE

To protect unlabeled traffic there is no need for dual label popping or "rL". Instead, all the repairing router needs to do when it detects that "pNH" is no longer reachable is to re-tunnel the packet towards "rNH" in a regular LSP

The next section presents the solution in an IP core.

3. Overview of the BGP FRR using Vector Labels in an IP Core

This section describes the BGP FRR using vector labels solution in an IP core for both labeled (AFI/SAFI 1/4, 2/4, 1/128, and 2/128) and unlabeled (AFI/SAFI 1/1, 2/1, 1/2, and 2/2) protected prefixes.

The primary difference between a MPLS core and an IP core is that the tunnels between edge routers are IP based such as [5][6][7]. In this section, we propose two alternatives: A completely pure IP core and a hybrid IP/MPLS core

3.1. Pure IP Core

In this section, we propose a scheme by which core routers are incapable of handling any kind of MPLS labels.

3.1.1. Control Plane

The pPE still needs to advertise the mapping (bgpNH,pNH) as in Section 2 but it does not allocate or advertise a vector label.

The rPE advertises rL with protected prefixes to all its iBGP peer as in MPLS core solution in Section 2.

Assume iPE decides that rPE is the repair PE for a protected prefix.

- o iPE pushes the usual VPN label for labeled prefixes
- o iPE pushes the repair label "rL" advertised by the chosen rPE
- o iPE pushes *two* IP tunnel headers on the packet
 - o Repair tunnel header. This will be the inner tunnel header with destination address rNH towards the rPE
 - o Protected tunnel header: This will be the outer tunnel header with destination address pNH towards the pPE

3.1.2. Forwarding plane during Steady State (when pPE is reachable)

1. iPE pushes the VPN label and the repair label followed by the two tunnel headers described in the previous section
2. rP: No special behavior necessary
3. pPE
 - a. Decapsulates *two* tunnel headers and the repair label "rL"
 - b. Uses the contents of the packet underneath

3.1.3. Forwarding plane at Failure (when pPE is unreachable)

1. iPE is not yet aware of the failure so its behavior remains the unchanged.

2. rP

- a. Decapsulates the outer tunnel header towards pNH
- b. Uses the repair tunnel header to forward the packet towards rPE

3. rPE

- a. Decapsulates the tunnel header
- b. Uses the repair label "rL" to forward the packet to the correct CE
 - i. Pop rL and the VPN label under it
 - ii. Use the forward the packet to the correct CE

3.2. Hybrid IP core

In this section, we assume that rP is capable of handling MPLS labels

3.2.1. Control Plane

The pPE needs to advertise the mapping (bgpNH,pNH). iPE also needs to allocate a vector label for each known rPE and advertise the mapping (pNH,rNH,vL) to all its iBGP peers and to candidate repair core routers. This behavior is identical to iPE behavior in MPLS core in Section 2.

The rPE advertises rL with protected prefixes to all its iBGP peer as in the case of MPLS core described in Section 2.

Assume iPE decides that rPE is the repair PE for a given prefix:

- o iPE pushes the usual VPN label for labeled prefix
- o iPE pushes the repair label "rL" advertised by the chosen rPE
- o Pushes the vector label of the chosen rPE
- o iPE pushes two the protected tunnel header: This will be the outer IP tunnel header with destination address pNH towards the pPE

rP behavior is identical to its behavior in an MPLS core in Section 2.

3.2.2. Forwarding Plane during Steady State (when pPE is reachable)

4. iPE pushes the two tunnel headers described in the previous section
5. rP: No special behavior necessary
6. pPE
 - a. Decapsulates the outer tunnel headers plus **two** labels (vL and rL
 - b. Uses the contents of the packet after decapsulation to forward the packet

3.2.3. Forwarding plane at Failure (when pPE is unreachable)

7. iPE is not yet aware of the failure so its behavior remains the same
8. rP
 - a. Decapsulates the tunnel header towards pNH
 - b. Pops the vector label "vL"
 - c. Looks up the vector label "vL" in the label context identified by pNH. The lookup should yield the rNH
 - d. Encapsulates the packet into a tunnel header with destination address rNH and forwards the packet towards rPE
9. rPE
 - a. Decapsulates the tunnel header
 - b. Uses the repair label "rL" to forward the packet to the correct CE
 - i. Pop **two** labels for labeled traffic
 - ii. Forward the packet to the correct CE

4. Rules for Choosing and Managing the Repair path

This section specifies rules governing how a protectable edge router pPE chooses and advertises the repair path. Other than the rules in this section, the method of choosing the repair path is beyond the scope of this document.

4.1. General Rules for Managing the Repair Path

This section specifies general rules for choosing the repair path for both labeled and unlabeled prefixes.

1. A repair PE MUST be another edge router that advertises the same prefix to the protected edge router pPE via IBGP peering.
2. If a repairing P router "rP" determines that the path taken by the repair tunnel to a repair edge router rPE passes through the protected edge router pPE, then the repairing router "P" MUST NOT install this repair path in its forwarding plane. Instead, the repairing "p" router MAY use other paths that do not pass through pPE or use existing core FRR mechanisms such as [12], [13], and [14].
3. Let the protected next-hop pNH match the IGP route pR. If the "rP" determines that the repair tunnel to a repair edge router passes through a next-hop of the IGP route pR, then the repairing router SHOULD NOT install this repair path in its forwarding plane.
4. A protected next-hop uniquely identifies an protected PE within a BGP-free core. Thus a protected next-hop NH MUST NOT be advertised by two different pPEs.
5. At any point in time, for the same primary and repair next-hops pNH and rNH, only one advertisement is valid. Thus for the same value of pNH and rNH, an advertisement of the pair (pNH,rNH) MUST override or be preceded by the withdrawal of any previously advertised pair (pNH,rNH).
6. If the repair PE "rPE" advertises one or more protected next-hops, then the repair next-hop "rNH" MUST be different from any protected next-hop "pNH" advertised by rPE

If rules (1), (2), and (3), then the tunnel to the repair edge router rPE does not provide protection against the failure of the edge node ePE. Instead it provides core protection against the failure of the path through the core leading to the protected edge node pPE. Thus existing core FRR protection mechanisms such as those specified in [12], [13], and [14] can be used instead.

Rules (4), (5), and (6) ensures that there is no ambiguity about the primary and repair next-hops

4.2. Rules for Choosing the Repair Path for Labeled Prefixes

This section specifies rules in additions to those mentioned in Section 4.1 by which an edge router iPE chooses and advertises the repair path for a protected labeled prefix P/m.

A edge router iPE MUST only choose the edge router rPE and the underlying repair label rL as a repair path for the prefix P/m if the "rL" allocated on per-VPN or per-CE/per-next-hop basis.

The reason for this rule is that "rL" is advertised as path attributes in MP/BGP updates. If "rL" is allocated on per-prefix basis, then attribute packing will be severely impacted

5. Inter-operability with Existing IP FRR Mechanisms

Current existing IP FRR mechanisms can be divided into two categories: core protection and edge protection. Core protection techniques, such as [12], [13], and [14], provide protection against internal node and/or link failure. Thus the technique proposed in this document is not related to existing IP FRR mechanisms. If the failure of an internal node or link results in completely disconnecting a protectable edge node, then an administrator MAY configure the repairing router to prefer the technique proposed in this document over existing IP FRR mechanisms.

Edge protection techniques, such as [16] provide protection against the failure of the link between PE and CE routers. Thus existing PE-CE link protection can co-exist with the techniques proposed in this document because the two techniques are independent of each other.

6. Example

We will use an LDP core as an example. Consider the diagram depicted in Figure 2 below. We assume that the PEs advertise repair labels as specified in [15]

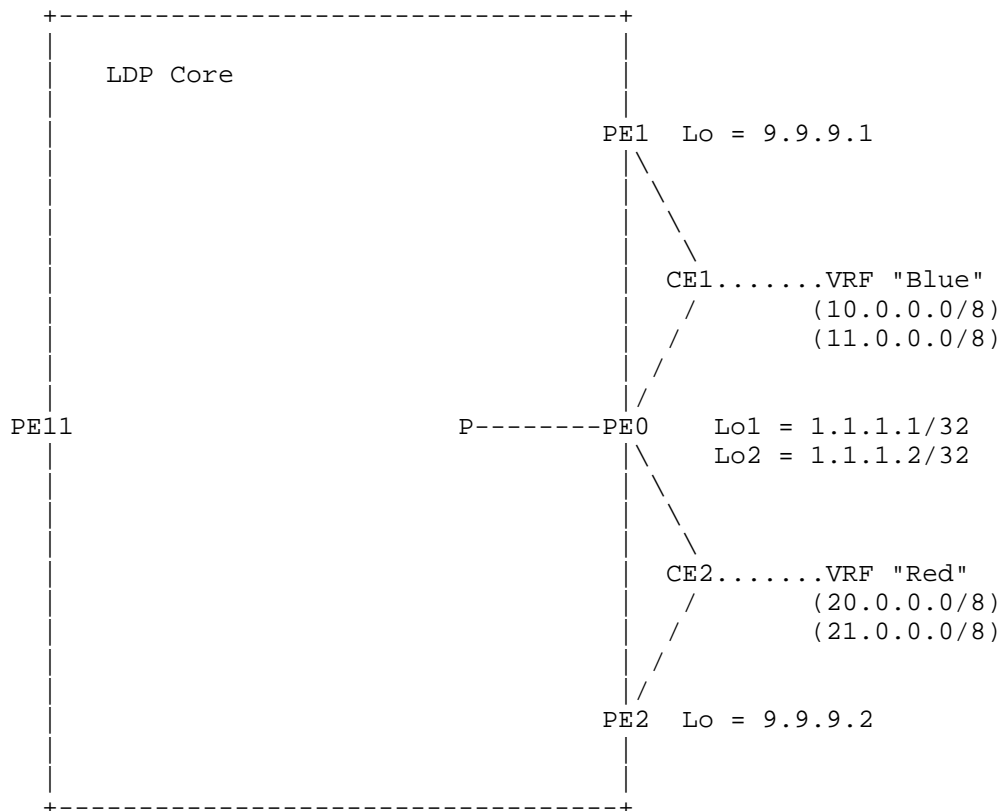


Figure 2 : Edge node BGP FRR in LDP core

- o In Figure 2, PE0 is the pPE for VRFs "Blue" and "Red". PE1 and PE2 are the rPEs for VRFs "Blue" and "Red", respectively. VRF Blue has 10.0.0.0/8 and 11.0.0.0/8 and VRF Red has 20.0.0.0/8 and 21.0.0.0/8
- o Assuming PE0 uses per prefix label allocation, PE0 assigns the VPN labels 4100, 4200, 4300, and 4400 to 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 respectively. PE0 advertises the prefixes 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 using MP/BGP as usual

6.1. Control Plane

1. rPEs Allocate and advertise Repair labels

- a. Acting as a rPE, PE1 allocates (on per-CE basis) and advertises a repair label rL1=3100 with the prefixes 10.0.0.0/8 and 11.0.0.0/8 to all iBGP peers
- b. Similarly, PE2 allocates and advertises the repair label rL2=3200 with the prefixes 20.0.0.0/8 and 21.0.0.0/8

2. pPE calculates and advertises the pNH

- a. Assume that PE0 uses "Loopback0" as the BGP next-hop, PE0 automatically picks Loopback2 as the pNH. As such PE0 advertises (bgpNH,pNH)=(1.1.1.1,1.1.1.2) to all iBGP peers including the iPE PE11.
- b. When the iPE "PE11" receives (bgpNH,pNH)=(1.1.1.1,1.1.1.2), PE11 understands that if it wants to protect traffic whose bgpNH=1.1.1.1 against the failure of the node 1.1.1.1, PE11 has to tunnel the traffic to 1.1.1.2 instead of 1.1.1.1

3. pPE allocates and advertizes vector labels

- a. On receiving the repair labels 3100 and 3200 from PE1 and PE2, respectively, PE0 detects that there are two rPEs: PE1 and PE2. AS such PE0 assigns two vector labels vL1 = 1100 and vL2 = 1200 to PE1 and PE2, respectively
- b. PE0 advertises (1.1.1.2, 9.9.9.1, 1100) and (1.1.1.2, 9.9.9.2, 1200) to all iBGP peers, including the ingress PE PE11
- c. On receiving (1.1.1.2, 9.9.9.1, 1100) and (1.1.1.2, 9.9.9.2, 1200), the ingress PE PE11 understand that if it were to pick 9.9.9.1 as the rPE for packet tunneled to 1.1.1.2, then it has to push the vector label 1100. Similarly, to protect a packet tunneled to 1.1.1.2 using the rPE 9.9.9.2, then it has to push the vector label 1200.
- d. PE0 also advertises (1.1.1.2, 9.9.9.1, 1100) and (1.1.1.2, 9.9.9.2, 1200) to all candidate repairing core routes, including the core router "P".

4. The repairing core router creates the repair state

- a. Acting as a rP, the core router "P" receives the advertisements (1.1.1.2, 9.9.9.1, 1100) and (1.1.1.2, 9.9.9.2, 1200) from PE0.
- b. rP understands that it has to pop *3* labels when it receives a packet whose top label is the LDP label for 1.1.1.2
- c. rP creates a label context identified by the LDP label of 1.1.1.2/32
- d. rP inserts the following two label entries in the created label context
 - i. 1100-->9.9.9.1
 - ii. 1200-->9.9.9.2

5. The ingress PE calculates the rPEs

- a. PE11 receives an advertisement for 10.0.0.0/8, 11.0.0.0/8, 20.0.0.0/8, and 21.0.0.0/8 from PE0 with the BGP next-hop=1.1.1.1. Because PE11 received (bgpNH,pNH)=(1.1.1.1,1.1.1.2) from PE0, then PE11 knows that to protect traffic tunneled to PE0, it has to tunnel the traffic to 1.1.1.2 instead of 1.1.1.1
- b. PE11 receives an advertisement from PE1 for 10.0.0.0/8 and 11.0.0.0/8 with the repair label 3100
- c. Hence PE11 picks PE1 as the rPE for the prefixes 10.0.0.0/8 and 11.0.0.0/8 with rNH=9.9.9.1 and rL=3100. Remember that the vector label for 9.9.9.1 is 1100.
- d. Similarly, PE11 receives an advertisement from PE2 for 20.0.0.0/8 and 21.0.0.0/8 with the repair label 3200
- e. Hence PE11 picks PE2 as the rPE for the prefixes 10.0.0.0/8 and 11.0.0.0/8 with rNH=9.9.9.2 and rL=1200. Remember that the vector label for 9.9.9.1 is 1100.

6.2. Forwarding Plane at Steady State (When PE0 is reachable)

1. Ingress PE PE11

- a. Traffic for VRF "Blue"
 - i. PE11 receives a packet for VRF Blue with destination address 10.1.1.1 from an external router.

- ii. PE11 pushes the following labels
 - 1. The VPN label 4100
 - 2. The Repair label 3100
 - 3. The vector label 1100
 - 4. The LDP label for 1.1.1.2
- b. Traffic for VRF "Red"
 - i. PE11 receives a packet for VRF Red with destination address 20.1.1.1 from an external router
 - ii. PE11 pushes the following labels
 - 1. The VPN label 4300
 - 2. The Repair label 3200
 - 3. The vector label 1200
 - 4. The LDP label for 1.1.1.2
- 2. Penultimate Hop of PE0 (Which is also the rP "P")
 - a. Receives a packet with top label for the protected next-hop 1.1.1.2
 - b. Pops *3* labels
 - c. Forwards the packet to 1.1.1.2
- 3. Protected PE PE0
 - a. Traffic for VRF "Blue"
 - i. PE0 receives traffic with the top label 4100.
 - ii. 4100 is the VPN label for VRF "Blue"
 - iii. PE0 pops the label 4100 and forwards the packet to CE1
 - b. Traffic for VRF "Red"
 - i. PE0 receives traffic with the top label 4300.
 - ii. 4300 is the VPN label for VRF "Red"

iii. PE0 pops the label 4300 and forwards the packet to CE2

6.3. Forwarding Plane at Failure (When PE0 is not reachable)

1. The ingress PE PE1

Does not know about the failure yet and hence it does not change its behavior.

2. Repair PE rP

a. Traffic for VRF "Blue"

- i. Receives a packet with the top label being the LDP label for 1.1.1.2
- ii. 1.1.1.2 is not reachable
- iii. Pop the LDP label of 1.1.1.2. The vector label 1100 is under it
- iv. Lookup the vector 1100 in the label context of 1.1.1.2. The lookup yields the LDP label of the rNH 9.9.9.1
- v. Swap the vector label 1100 with the LDP label of the of 9.9.9.1 and forward the packet towards PE1

b. Traffic for VRF "Red"

- i. Receives a packet with the top label being the LDP label for 1.1.1.2
- ii. 1.1.1.2 is not reachable
- iii. Pop the LDP label of 1.1.1.2. The vector label 1200 is under it
- iv. Lookup the vector 1200 in the label context of 1.1.1.2. The lookup yields the LDP label of the rNH 9.9.9.2
- v. Swap the vector label 1200 with the LDP label of the of 9.9.9.2 and forward the packet towards PE2

3. The repair Router "PE1"

- a. The penultimate hop of PE1 performs the usual penultimate hop popping

- b. PE1 receives a packet with the top label equals the repair label 3100, which was allocated on per-CE basis and points to CE1

- c. PE1 pops *2* labels and forwards the packet to CE1

4. The repair Router "PE2"

- a. The penultimate hop of PE2 performs the usual penultimate hop popping

- b. PE1 receives a packet with the top label equals the repair label 3200, which was allocated on per-CE basis and points to CE2

- c. PE2 pops *2* labels and forwards the packet to CE2

7. Security Considerations

No additional security risk is introduced by using the mechanisms proposed in this document

8. IANA Considerations

No requirements for IANA

9. Conclusions

This document proposes a method that allows fast re-route protection against edge node failure or complete disconnected from the core in a BGP-free core. The method proposed has the following advantages

- o Very scalable:

- o No router has to copy the routing table of another router

- o Minimum additional prefixes injected in the core. In fact, at most one additional prefix per pPE is injected and only if there is no spare IP address on the pPE

- o Minimal provisioning overhead:

- o If there is a spare IP address on the pPE, then the provisioning effort is just enablement. If not, then the provisioning effort is just to configure a distinct IP address on each pPE to act as the pNH.

- o Absolutely no restriction on which PE is connected to which VRF.
- o On a PE where BGP FRR is already configured, moving, connecting, or disconnecting a CE to/from the PE requires zero operator intervention to protect prefixes.
- o Immunity to misconfiguration: the only configuration that may be required is a distinct pNH on each pPE. The mapping (bgpnh,pPE) and (pNH,rNH,vL) is advertised to all BGP peers. If the operator configures the same pNH on two different pPE, then the misconfiguration will be detected almost immediately
- o No Need for IP or TE FRR: Because the exit point of the repair tunnel from rP to rPE is different from the primary tunnel exit point
- o Works in both MPLS core and IP core
- o Works with per-CE, per-VRF and per-prefix label allocation
- o Can be incrementally deployed. There is no flag day. Different routers can be upgraded at different times
- o Zero impact on the paths taken by traffic: Enabling/deploying the feature described in this document has no effect on the paths taken by traffic at steady state

10. References

10.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006
- [3] Bates, T., Chandra, R., Katz, D., and Rekhter Y., "Multiprotocol Extensions for BGP", RFC 4760, January 2007
- [4] Malhotra, P. and Rosen, E., "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009
- [5] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.

- [6] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.
- [7] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

10.2. Informative References

- [8] Marques, P., Fernando, R., Chen, E., Mohapatra, P., Gredler, H., "Advertisement of the best external route in BGP", draft-ietf-idr-best-external-04.txt, April 2011.
- [9] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.
- [10] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [11] De Clercq, J., Ooms, D., Prevost, S., Le Faucheur, F., "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, February 2007
- [12] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.
- [13] Shand, S., and Bryant, S., "IP Fast Reroute", RFC5714, January 2010
- [14] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.
- [15] Bashandy, A., Pithawala, P., and Heitz, J., "Scalable, Loop-Free BGP FRR using Repair Label", draft-bashandy-idr-bgp-repair-label-02.txt, July 2011
- [16] O. Bonaventure, C. Filss, and P. Francois. "Achieving sub-50 milliseconds recovery upon bgp peering link failures," IEEE/ACM Transactions on Networking, 15(5):1123-1135, 2007

11. Acknowledgments

Special thanks to Clarence Filss, Eric Rosen, Stewart Bryant, and Pradosh Malhotra for the valuable comments

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A. Other Algorithms to Allocate and Disseminate Vector Labels

This section outlines two alternate algorithms for Allocating and distributing vector label "vL" to "rPE" mapping. The alternate algorithms can be divided into two categories, iPE chooses the repair path and pPE chooses the repair path

A.1. iPE chooses the repair path

A.1.1. Allocating Vector Labels using a Hash Function

In the method of allocating and advertising vector labels outlined in Sections 2 and 3 each pPE allocates and binds a vector label to each known rPE. As a result, the same rPE may be bound to multiple vector labels by multiple pPEs and thus requiring additional storage on the rP. In this section, we propose a method by which a vector label is computed using a hash function based on the numerical value of rNH

A.1.1.1.1. Calculating and distributing the mapping rNH->vL to different routers

1. We assume that all routers in a BGP free core, including edge router, agree on the set of candidate repair next-hops. This can be achieved via default behavior (e.g. all host routes) or some sort of configuration, such as ISIS administrative tags
2. No need for pPE to advertise the (pNH,rNH,vL) to iPE or rP
3. Each candidate rP and iPE calculates the vL for each candidate repair next-hop rNH
4. The rP inserts the calculated mapping vL-->rNH in a "repair label context" that is common for all protected PEs instead of having separate label context for each pPE.
5. If iPE chooses rNH as the repair next-hop for traffic tunneled to pNH, iPE calculates the vL corresponding to the chosen rNH and pushes vL as described in Sections 2.1.
6. On pPE failure, the lookup for vL occurs in the common "repair label context"

IN the next subsections, we outline two risks of using the hash function for rNH-->vL mapping.

A.1.1.1.2. Risk of Mis-configuration leading to Mismatch in rNH-->vL Mapping

1. Due to misconfiguration, some routers may not have the identical sets of candidate repair next-hops "rNH's" or use the same hash function to calculate vL. For example, an upgraded router may have a new hash function enabled or the ISIS administrative tags may not be associated with all candidate rNHs
2. To alleviate this risk, we propose that each rPE associates the calculated value of vL for each rNH in an optional TLV in IGP
3. If a router finds that its calculated value for rNH-->vL is different from the value received from the corresponding rPE, then the router can raise an alarm,

A.1.1.1.3. Risk of forwarding to Incorrect VRF during convergence only

Identical mapping of rNH-->vL is only guaranteed if the set of candidate rNH is the same on all routers. Because each router calculates rNH-->vL independently, there is a minor risk of forwarding to incorrect VRF. Consider the following example

1. The risk exists even if every rPE advertises the vL of its own rNH
2. Two rNH's, say rNH1 and rNH2, map to the same vL, even if rNH1, and rNH2 protect different prefixes
3. rPE1 and rPE2 have not yet heard each other mappings
4. iPE learns about vL-->rNH1 before vL-->rNH2
5. rP/PLR learns about vL-->rNH2 before vL-->rNH1
6. If pPE fails during the short period before iPE and rP can detect the vL collision, rP re-routes traffic to rNH2 but the repair label pushed by iPE is for rNH1.

A.1.2. pPE Allocates and advertises vL with protected prefixes

1. pPE allocates a single vL for all prefixes reachable via the same CE. If two prefixes bound to the same vL are protected by different rPE's, then pPE MUST re-advertise the second protectable prefix with a different vL to all ingress PEs
2. pPE always advertises (pNH, vL) with protected prefixes as optional attributes all the time even if there is no rPE

3. iPEs and the pPE agree on the way to pick the rPE. E.g. if there are multiple rPEs, choose the one with lowest router ID
4. When rPE advertises rL for a protected prefix
 - a. Both pPE and iPE will get the update
 - b. Both pPE and iPE will choose the same rPE for the protected prefix
5. iPE associate the correct triplet (pNH, vL, rL) with protected prefixes without getting a re-advertisement for the prefix from pPE
6. pPE Informs about (pNH, rNH, vL)
7. Hence rP/PLR knows that the vector label vL maps to rNH in the label context of pNH.
8. rP/PLR inserts vL-->rNH in the context of pNH

A.1.2.1.1. Risk of forward to Incorrect VRF during Convergence Only

The conditions for the risk to exist

- o More than one rNH, say rNH1 and rNH2, protect the same prefix
- o iPE learns about rNH1 and has not yet learnt about rNH2
- o pPE learns about rNH2 and has not yet learnt about rNH1
- o pPE fails during this time period

How incorrect forwarding can occur

- o pPE maps vL to rPE2 on rP/PLR while iPE maps vL to rPE1
- o pPE fails during this short period,
- o rP/PLR re-routes the packet to rPE2 but the repair label pushed by iPE belongs to rPE1 (say rL1)

A.2. pPE chooses rPE and distributes the mapping of vL-->rNH

In Sections 2, 3, and A.1 the ingress PE chooses the rPE for every protectable prefix. While it causes less churn because there is never a need to re-advertise protected prefixes, it is difficult to

configure a policy to control the choice of the rPE if the policy has to be applied to all iPEs. In this Section, we propose an algorithm to select rPE and advertise vL-->rNH via pPE instead of iPE

1. pPE allocates a single vL for all prefixes reachable via the same CE
 - a. We assume that prefixes reachable via the same CE or belong to the same VRF are protectable by the same rPE
 - b. If two prefixes bound to the same vL are protected by different rPE's, then pPE MUST re-advertise the second protectable prefix with a different vL to all ingress PEs
2. pPE always advertises (pNH, vL) with protected prefixes as optional attributes all the time even if there is no rPE. Remember that pNH,vL means the vector label for protected traffic tunneled to pNH is vL
3. Based on rPE advertisement, pPE decides that the repair next-hop for a given protected prefix P/m is rNH. pPE sends the mapping (vL,rNH) similar to [4] as a separate advertisement to iPEs
4. Suppose two prefixes P1/m1 and P2/m2 are associated with the same vector label vL1 but are protected by two different repair PEs: rNH1 and rNH2
 - a. Re-advertise P2/m2 with a new vector label vL2
 - b. pPE sends the mapping(vL1,rNH1) and (vL2,rNH) in a separate advertisement to iPEs
 - c. The re-advertisement of the prefix p2/m2 with the new vector label vL2 must be done BEFORE sending the vector label mapping to guarantee correct forwarding

Unlike the schemes in Sections A.1.1 and A.1.2 there is no risk of forwarding to incorrect VRF because pPE is the only source of mapping vL-->rNH

A.3. Combination of iPE and pPE Choosing rPE

- o pPE can choose the rPE by specifying the mapping of vL to rNH, re-advertising/advertising the protected prefix with rNH, or a combination of both
- o pPE decides the prefixes for which it chooses the rPE based on various factors. For example

- o Option 1: The operator can configure the prefixes for which the pPE can choose the rPE.
- o Option 2: If there is more than one rPE, then pPE chooses the rPE. Otherwise, it is left to iPE
- o There are probably other options
- o As long as the pPE does not specify the rPE for a prefix, then the iPE is free to choose the rPE, otherwise, the iPE has to abide by pPE choice

A combination of iPE and pPE choosing the rPE reduces the provisioning overhead when configuring a policy to choose the rPE at the expense of increasing the churn.

Authors' Addresses

Ahmed Bashandy
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: bashandy@cisco.com

Nagendra Kumar
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: naikumar@cisco.com

Maciek Konstantynowicz
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134
Email: mkonstan@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: March 29, 2013

Camilo Cardona
Pierre Francois
IMDEA Networks
September 25, 2012

Making BGP filtering an habit: Impact on policies
draft-cardona-filtering-threats-00

Abstract

This draft describes potential threats to the Internet routing policies of an autonomous system due to filtering of more specific BGP prefixes by its neighboring domains.

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 29, 2013.

Copyright Notice

Copyright (c) 2012 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. Filtering overlapping prefixes | 3 |
| 2.1. Local filtering | 4 |
| 2.2. Remotely triggered filtering | 5 |
| 3. Uses of more specific prefix filtering that violate policies | 6 |
| 3.1. Violation caused by Local filtering | 7 |
| 3.1.1. Initial setup | 7 |
| 3.1.2. Violation of Policy - Case 1 | 8 |
| 3.1.3. Violation of Policy - Case 2 | 9 |
| 3.2. Violation caused by remotely triggered filtering | 10 |
| 3.2.1. Initial setup | 10 |
| 3.2.2. Injection of a more specific | 11 |
| 3.2.3. Limiting the scope of the more specific | 12 |
| 4. Techniques to detect dataplane-based policy violations | 14 |
| 4.1. Being the victim of the policy violation | 14 |
| 4.2. Being a contributor to the policy violation | 14 |
| 5. Techniques to counter policy violations | 15 |
| 5.1. Reactive counter-measures | 15 |
| 5.2. Anticipant counter-measures | 16 |
| 5.2.1. Neighbor-specific forwarding | 16 |
| 5.2.2. Access lists | 16 |
| 5.2.3. Automatic filtering | 16 |
| 6. Conclusions | 16 |
| 7. References | 17 |
| Authors' Addresses | 17 |

1. Introduction

It is common practice for network operators to propagate overlapping prefixes along with the prefixes that they originate. On the other hand, it can be beneficial for some Autonomous Systems (ASes) to filter overlapping prefixes (such operation needs to be translated into various requirements in order to be automatically performed) DRAFT-WHITE [1].

BGP makes independent, policy driven decisions for the selection of the best path to be used for a given IP prefix. However, in the data plane, the longest prefix match forwarding rule "precedes" the application of such policies. The existence of a prefix p' that is more specific than a prefix p in the Routing Information Base (RIB) will indeed let packets whose destination matches p' be forwarded according to the next hop selected as best for p' (the overlapping prefix). This process takes place by disregarding the policies applied in the control plane for the selection of the best next-hop for p (the covering prefix). When overlapping prefixes are filtered and packets are forwarded according to the covering prefix, the discrepancy in the routing policies applied both covering and overlapping prefixes can lead to a violation of policies of Internet Service Providing (ISPs) still holding a path towards the overlapping prefix.

This document presents examples of such potential threats, and discusses solutions to the problem. The objective of this draft is to enable the use of prefix filtering while making the routing community aware of the cases where the effects of filtering might turn to be negative for the business of ISPs.

The rest of the document is organized as follows: Section 2 describes some cases in which it is favorable for an AS to filter overlapping prefixes. In Section 3, we provide some scenarios in which the filtering of overlapping prefixes lead to policy violations of other ASes. Section 4 and Section 5 introduce some techniques that ASes can use for, respectively, detect and react to policy violations.

2. Filtering overlapping prefixes

There are different scenarios where filtering an overlapping prefix is relevant to the operations of an AS. In this section, we illustrate examples of these scenarios. We differentiate cases in which the filtering is performed locally from those where the filtering is triggered remotely, by using BGP communities. These scenarios will be used as a base in Section 3 for describing side effects bound with such practices, notably policy violations in the

ASes surrounding the AS applying the procedure.

2.1. Local filtering

Let us first analyze the scenario depicted in Figure 1. AS1 and AS2 are two large autonomous systems spanning a large geographical area and peering in 3 different physical locations. Let AS1 announce prefix 10.0.0.0/22 through the sessions established between the two ASes over all peering links. Additionally, let us define that there is part of AS1's network which exclusively uses prefix 10.0.0.0/24 and which is closer to one specific peering point than to others (right peering link). With the purpose of receiving the traffic from AS2 to prefix 10.0.0.0/24 on the right peering link, AS1 could announce the overlapping prefix on this specific peering point. At the time of the establishment of the peering, it can be defined by both ASes that hot potato routing would happen in both directions of traffic. In this scenario, it becomes relevant for AS2 to enforce such practice by detecting the described situations and automatically issue the appropriate filtering. In this case, by implementing these automatic procedures, AS2 would detect and filter prefix 10.0.0.0/24.

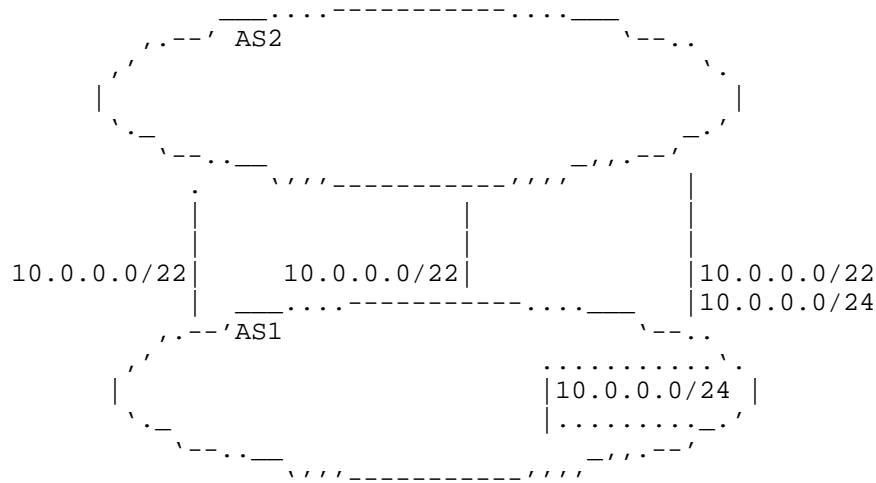


Figure 1: Basic scenario local filtering 1

There are other cases in which there could exist a need for local filtering. For example, a dual homed AS receiving an overlapping prefix from only one of its providers. Figure 2 depicts a simple example of this case.

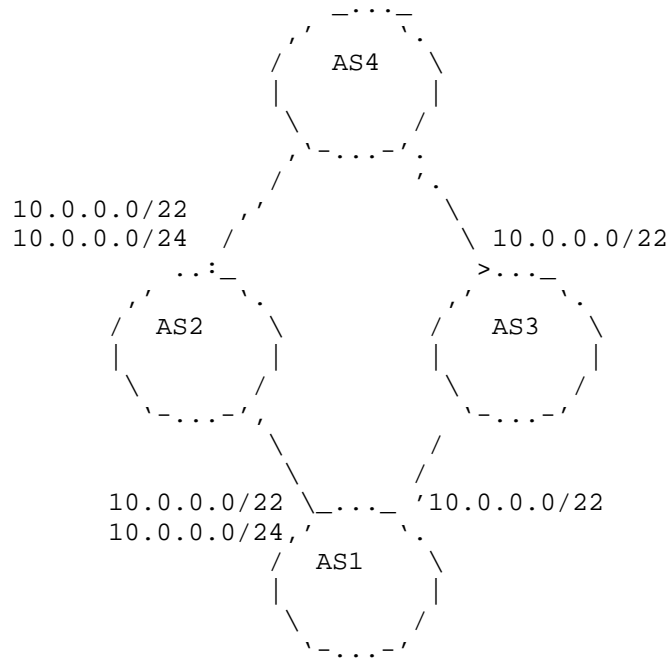


Figure 2: Basic scenario local filtering 2

In this scenario, prefix 10.0.0.0/22 is advertised by AS1 to AS2 and AS3. Both AS propagate the prefix to AS4. Additionally, AS1 advertises prefix 10.0.0.0/24 to AS3, which subsequently propagates the prefix to AS4. 10.0.0.0/22 is a covering prefix for 10.0.0.0/24.

It is possible that AS4 resolves to filter the more specific prefix 10.0.0.0/24. One potential motivation could be the economical preference of the path via AS2 over AS3. Another feasible reason is the existence of a technical policy by AS4 of aggregating incoming prefixes longer than /23.

The above examples illustrate two of the many motivations to configure routing within an AS with the aim of ignoring more specific routes. Operators have reported applying these filters in a manual fashion INIT7-RIPE63 [2]. The relevance of such practice led to investigate automated filtering procedures (DRAFT-WHITE [1]).

2.2. Remotely triggered filtering

ISPs can tag the BGP paths that they propagate to neighboring ASes with communities, so as to tweak the propagation behavior of the ASes

that handle such propagated paths [on_BGP_communities].

Some ISPs allow their direct and indirect customers to use such communities in order to let the receiving AS not export the path to some selected neighboring AS. By combining communities, the prefix could be advertised only to a given peer of the AS providing this feature. Figure 3 illustrates an example of this case.

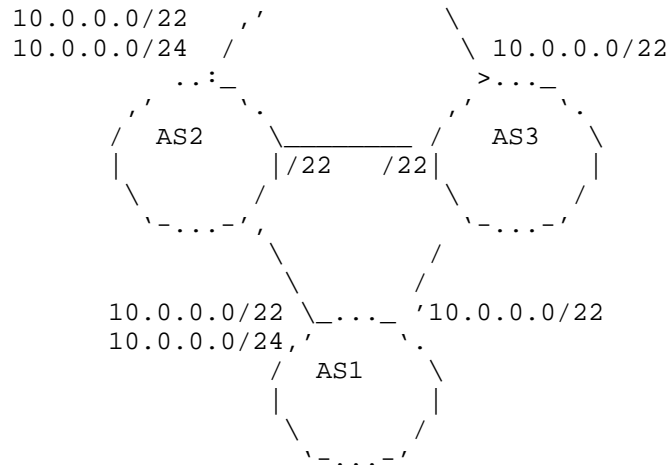


Figure 3: Remote triggered filtering

AS2 and AS3 are peers. Both ASes are providers of AS1. For traffic engineering purposes, AS1 could use communities to prevent AS2 from announcing prefix 10.0.0.0/24 to AS3.

Such technique is useful for operators to tweak routing decisions in order to align with complex transit policies. We will see in the later sections that by producing the same effect as filtering, they can also lead to policy violations at other, distant, ASes.

3. Uses of more specific prefix filtering that violate policies

We describe in this section three configuration scenarios which lead to the violation of the policies of an AS. Note that these examples do not capture all the cases where such policy violation can take place. More examples will be provided in the future revisions of this document.

3.1. Violation caused by Local filtering

In this section we describe cases in which an AS locally filters an overlapping prefix. We show how, depending on the situations of BGP policies, this decision leads to the violation of the policies of neighboring ASes.

3.1.1. Initial setup

We start by describing the basic scenario of this case in Figure 4.

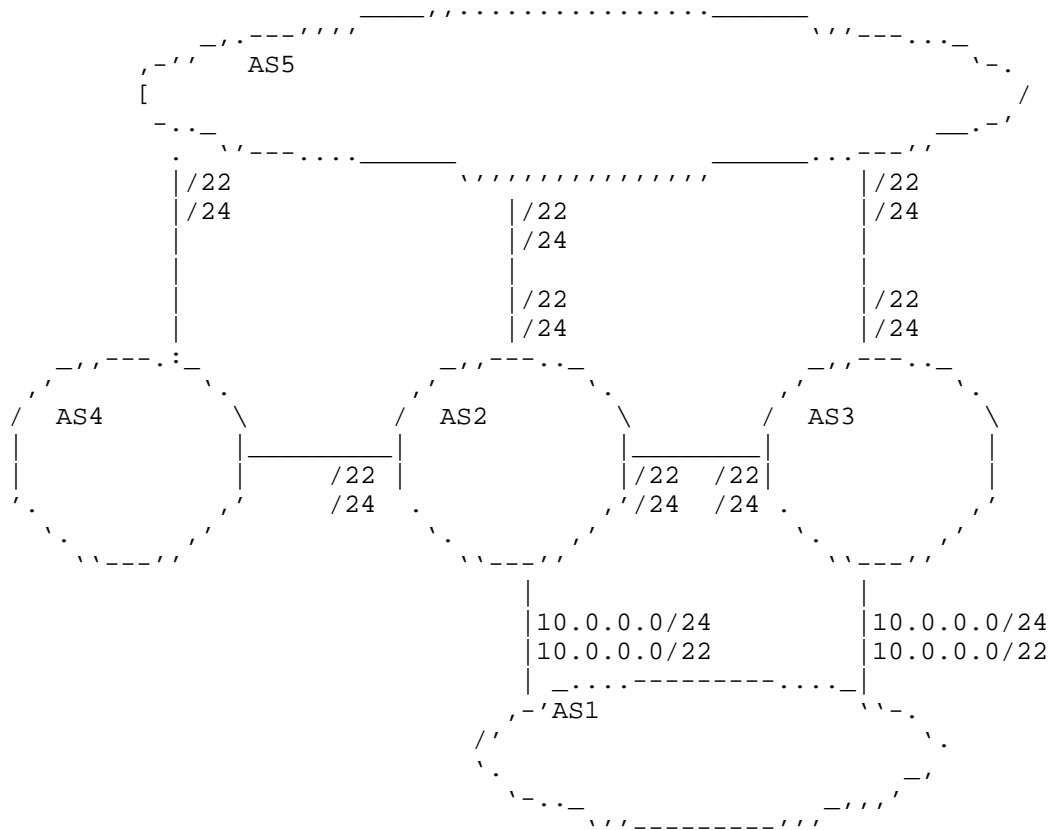


Figure 4: Initial Setup Local

AS1 is a customer of AS2 and AS3. AS2, AS3 and AS4 are customers of AS5. AS2 is establishing a free peering with AS3 and AS4. AS1 is

announcing a covering prefix, 10.0.0.0/22, and an overlapping prefix 10.0.0.0/24 to its providers. In the initial setup, AS2 and AS3 will announce the two prefixes to their peers and transit providers. AS4 receives both prefixes from its peer (AS2) and transit provider (AS5).

3.1.2. Violation of Policy - Case 1

In the next scenarios, we show that if AS4 filters the incoming overlapping prefix from AS5, there is a situation in which the policies of other ASes are violated.

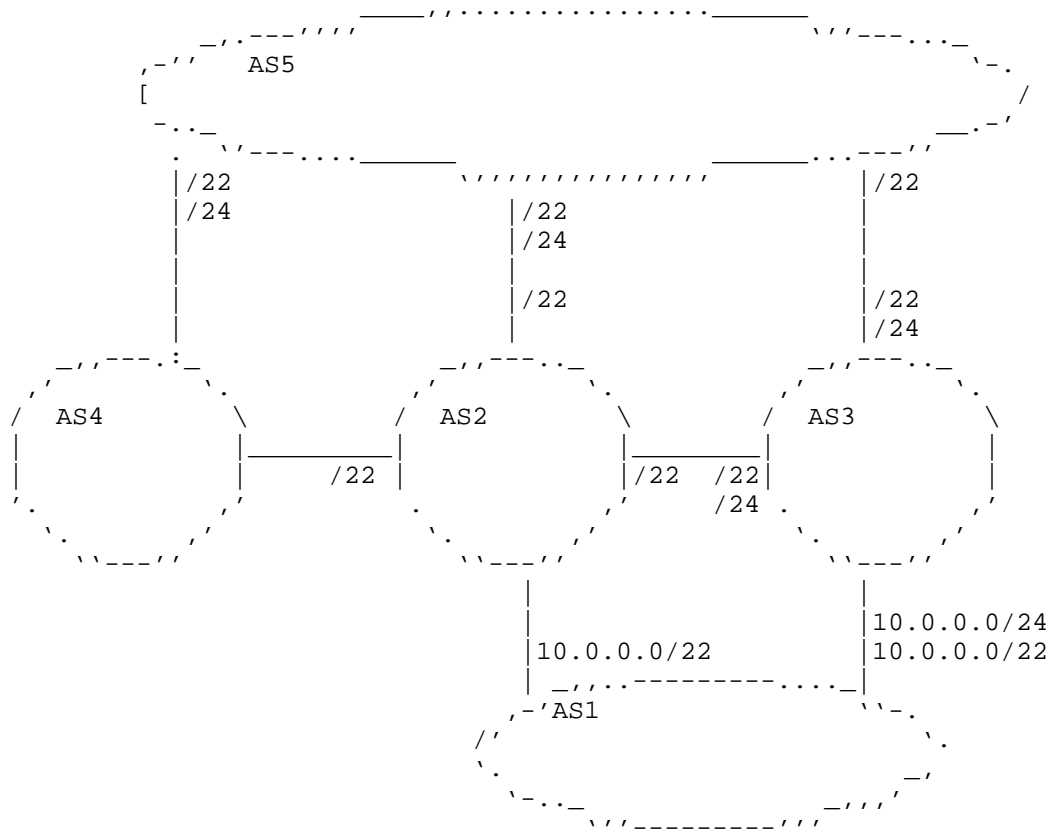


Figure 5: Initial Setup Local

Let us assume the scenario illustrated in Figure 5. For this case, AS1 only propagates the overlapping prefix to AS3. AS4 receives the overlapping prefix only from its transit provider, AS5.

The described example places AS4 in a situation in which it would be favorable for it to filter the announcement of prefix 10.0.0.0/24 from AS5. Subsequently, traffic originating from AS4 to prefix 10.0.0.0/24 is forwarded to AS2. As AS2 receives the more specific prefix from AS3, traffic originating from AS4 and heading to prefix 10.0.0.0/24 follows the path AS4-AS2-AS3-AS1. This violates the policy of AS2, since it forwards traffic from a peer to a non-customer neighbor.

3.1.3. Violation of Policy - Case 2

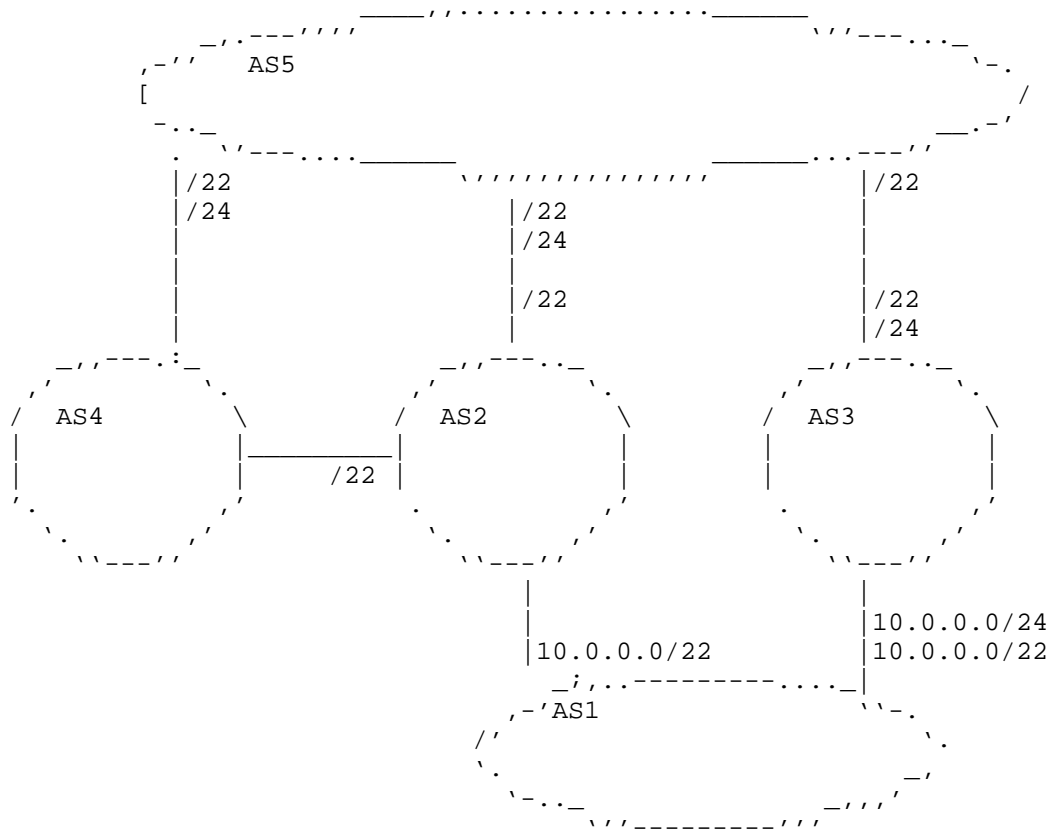


Figure 6: Initial Setup Local

Let us assume a second case where AS2 and AS3 are not peering and AS1

only propagates the overlapping prefix to AS3. AS4 receives the overlapping prefix only from its traffic provider, AS5. This case is illustrated in Figure 6.

Similar to the scenario described in Section 3.1.2, AS4 is in a situation in which it would be favorable to filter the announcement of prefix 10.0.0.0/24 from AS5. Subsequently, traffic originating from AS4 to prefix 10.0.0.0/24 is forwarded to AS2. Traffic originating in AS4 and heading for prefix 10.0.0.0/24 would follow the path AS4-AS2-AS5-AS3-AS1. This path violates the policy of AS2, as this AS is forwarding traffic from a peer to a transit network.

3.2. Violation caused by remotely triggered filtering

We present a configuration scenario in which an AS, using the mechanism described in Section 2.2, informs its provider to selectively announce a covering prefix, leading to the violation of a policy of another AS.

3.2.1. Initial setup

Let AS_cust be a customer AS of AS A and AS B. It owns 10.0.0.0/22, which it advertises through AS A and AS B. Additionally, AS A and AS B are peers.

Both AS A and AS B select their customer path as best, and propagate that path to their customers, providers, and peers.

Some remote ASes will route traffic destined to 10.0.0.0 through (... A Cust 10.0.0.0/24) while some others will route traffic along (... B Cust 10.0.0.0/24).

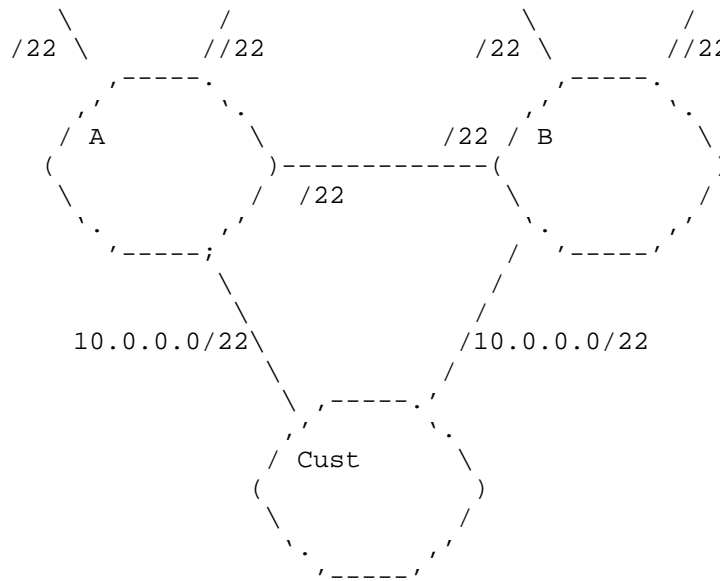


Figure 7: Example scenario

3.2.2. Injection of a more specific

Let AS_cust advertise 10.0.0.0/24 over AS B only. AS B propagates this prefix to its customers, provider and peers, including AS A.

From AS A's point of view, such a path is a "peer path", so that this path will only be advertised to its customers.

All ASes that are not in the customer branch of AS A will receive a path to the /24 that contains AS B, and not AS A, as AS A has not propagated the prefix to other ASes than its customers.

The ASes that are in the customer branch of AS A will receive a path to the /24 that contains AS B and AS A, as AS A has propagated that path to its customers. Some multi-homed customers of ISP A may also receive a path through ISP B, but not through ISP A, from other peering or provider links.

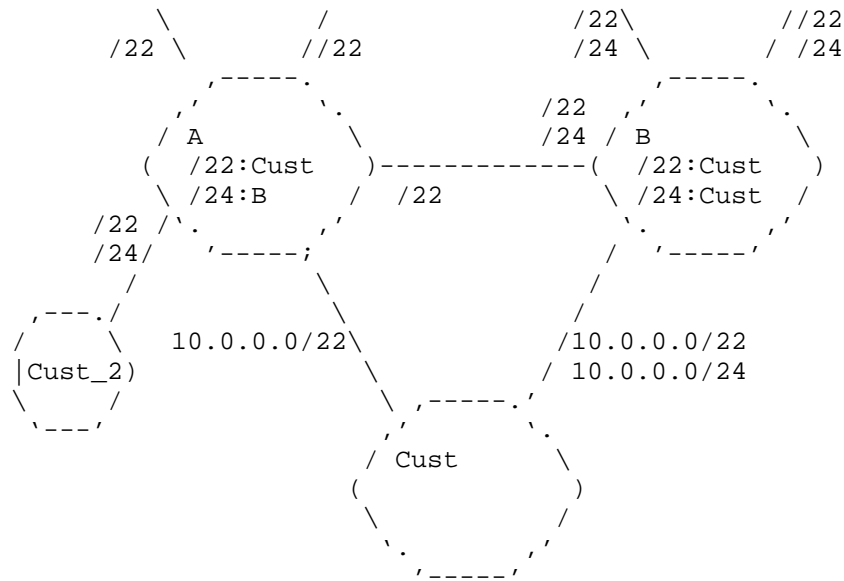


Figure 8: More Specific Injection

Any remote AS that is not lying in the customer branch of A, will receive a path for 10.0.0.0/24 through AS B and not through AS A.

Routing is consistent with usual Internet Routing Policies here, as AS A may only receive traffic destined to 10.0.0.0/24 from its customers, which it forwards to its peer AS B. AS B may receive traffic destined to 10.0.0.0/24 from its customers, providers, and peers, which it directly forwards to its customer AS Cust.

3.2.3. Limiting the scope of the more specific

Now, let us assume that 10.0.0.0/24, which is propagated by AS_Cust to AS B, is tagged so as to have AS B only propagate that path to AS A, using the techniques described in Section 2.2.

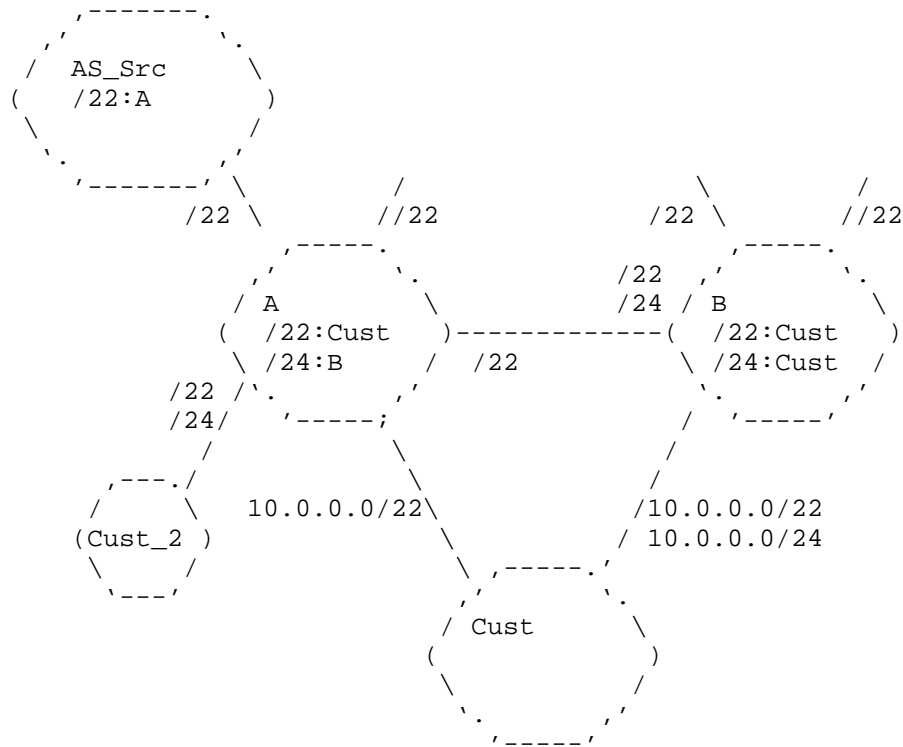


Figure 9: More Specific Injection

From AS A's point of view, such a path is a "peer path", so that this path will only be advertised by AS A to its customers.

All the ASes that are not in the customer branch of AS A nor in the customer branch of AS B will NOT receive a path to 10.0.0.0/24.

All these ASes will forward packets destined to 10.0.0.0/24 according to their routing state for 10.0.0.0/22.

Let us assume that AS_Src is such an AS, and that its best path towards 10.0.0.0/22 is through AS A. In that case, packets sent towards 10.0.0.1 by AS_Src will eventually reach AS A. However, in the dataplane of the nodes of AS A, the longest prefix match for 10.0.0.0 is 10.0.0.0/24, which is reached through AS B, a peer of AS A.

As AS_Src is by definition not in the customer branch of AS A, we are in a situation such that AS A is forwarding non customer originated

traffic along peering links, which violates its policies.

If the path towards 10.0.0.0/24 is propagated by B to its customers, the traffic originated by ASes in the customer branch of AS A will not follow policy-violating data-plane paths as the forwarding of traffic towards these destinations will always be based on FIB entries for 10.0.0.0/24. However, policy-violation can still take place for the traffic originated from all ASes that are neither in the customer branch of A nor in the customer branch of B.

4. Techniques to detect dataplane-based policy violations

We differentiate the techniques available for detecting policy violations from the cases in which the interested AS is the victim or contributor of such operations.

4.1. Being the victim of the policy violation

To detect that its policies have been violated, one ISP can monitor its NetFlow data so as to see if flows entering the ISP network through a non-customer link is being forwarded to a non-customer nexthop.

Detecting such a violation can be done by looking at BGP data to see whether there exists in the RIB a prefix P/p' more specific than P/p such that the nexthop for P/p' is through a peer (or a provider) while P/p is routed through a customer. For each such couple of prefixes, direct communication or looking glasses can be used in order to check whether non-customer neighboring ASes are propagating a path towards P/p (and not towards P/p') to their own customers, peers, or providers. This should trigger a warning as this would mean that ASes in the surrounding area of the current AS are forwarding packets based on the routing entry for the less specific prefix only.

4.2. Being a contributor to the policy violation

It can be considered as problematic to be a contributor of the policy violation as it appears as an abuse of other's network resources.

There may be justifiable reasons for one ISP to perform filtering, either to enforce establishing policies or to provide prefix advertisement scoping features to its customers. These can vary from trouble-shooting purposes to business relationships implementations. Restricting such features for the sake of avoiding contributing to potential policy violations in a peer's network is a bad option.

Netflow data does not help an ISP to detect that it is acting as a contributor of the policy violation. It is thus advisable to obtain as much information as possible of the Internet environment of the AS and assessing the risks of filtering of overlapping prefixes before implementing them.

Monitoring the manipulation of the communities that implement the scoping of prefixes in one's network is recommended to the ISPs which provide these features. The monitored behavior should then be faced against their terms of use.

5. Techniques to counter policy violations

Network Operators can adopt different approaches with respect to policy violation. We classify these actions according to whether they are anticipant or reactive.

Reactive approaches are those in which the operator tries to detect the situations and solves the policy violation through other means than using the routing system.

Anticipant or preventive approaches are those in which the routing system will not let the policy violation actually take place when the configuration scenario is set up.

5.1. Reactive counter-measures

An operator who detects that its policies have been violated can contact the ASes that are likely to have performed the propagation tweaks so as to have them change their behavior.

An operator can account the amount of traffic that has been subject to policy violation, and charge the peer that received the policy-violating traffic. That is, the operator can claim that it has been a provider of that peer for that part of the traffic that transited between the two ASes.

An operator can decide to filter-out the concerned more specific prefix at the peering session over which it was received. In the example of Figure 9, AS A would filter out 10.0.0.0/24 in its eBGP in-filter associated with the eBGP session with AS B. As a result, the traffic destined to that /24 would be forwarded by AS A along its link with AS_Cust, despite the actions performed by AS_Cust to have this traffic coming in through its link with AS B.

5.2. Anticipant counter-measures

5.2.1. Neighbor-specific forwarding

An operator can technically ensure that the traffic destined to a given prefix will be forwarded from an entry point of its AS, only on the basis of the set of paths that have been advertised over that entry point.

5.2.2. Access lists

An operator can configure its routers so as to have them dynamically install an access-list made of the prefixes towards which the forwarding of traffic from that interface would lead to a policy violation. Note that this technique actually lets packets destined to a valid prefix be dropped while they are sent from a neighboring AS that cannot know about the policy violation and hence had no means to avoid the policy violation.

In the example of Figure 9, AS A would install an access-list denying packets matching 10.0.0.0/24 associated with the interface connecting AS_Src. As a result, the traffic destined to that /24 would be dropped, despite the existence of a non policy-violating route towards 10.0.0.0/22.

5.2.3. Automatic filtering

As described in Section 3, filtering of overlapping prefixes can in some scenarios lead to policy violations. Nevertheless, depending on the autonomous system implementing such practice, this operation can in fact prevent these cases. This can be illustrated using the example described in Section 3.1.3: In Figure 6, if AS2 or AS3 filter prefix 10.0.0.0/24, there would be no policy violation for AS2.

6. Conclusions

In this document we described potential threats to policy violation of autonomous systems caused by the filtering of overlapping prefixes by external networks. We provide examples of scenarios of policy violations caused by these practices and introduce some techniques for their detection and counter. We observe that there are reasonable situations in which ASes could filter overlapping prefixes, however, we encourage that network operators implement this type of filters only after considering such threats.

7. References

[on_BGP_communities]

Donnet, B. and O. Bonaventure, "On BGP Communities", ACM SIGCOMM Computer Communication Review vol. 38, no. 2, pp. 55-59, April 2008.

[1] <<http://tools.ietf.org/html/draft-white-grow-overlapping-routes-00>>

[2] <<http://ripe63.ripe.net/presentations/48-How-more-specifics-increase-your-transit-bill-v0.2.pdf>>

Authors' Addresses

Juan Camilo Cardona
IMDEA Networks
Avenida del Mar Mediterraneo
Leganes 28919
Spain

Email: juancamilo.cardona@imdea.org

Pierre Francois
IMDEA Networks
Avenida del Mar Mediterraneo
Leganes 28919
Spain

Email: pierre.francois@imdea.org

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 18, 2013

J. Dong
M. Chen
Huawei Technologies
October 15, 2012

Distribution of MPLS Traffic Engineering (TE) LSP State using BGP
draft-dong-idr-te-lsp-distribution-00

Abstract

This document describes a mechanism to collect the Traffic Engineering (TE) LSP information using BGP. Such information can be used by external components for path reoptimization, service placement and network visualization.

Requirements Language

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

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <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 18, 2013.

Copyright Notice

Copyright (c) 2012 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. Carrying LSP State Information in BGP | 4 |
| 2.1. LSP Information NLRI | 4 |
| 2.2. LSP State Attribute | 6 |
| 3. IANA Considerations | 7 |
| 4. Security Considerations | 7 |
| 5. Acknowledgements | 7 |
| 6. References | 7 |
| 6.1. Normative References | 7 |
| 6.2. Informative References | 8 |
| Authors' Addresses | 8 |

1. Introduction

In some network environments, the states of established Multi-Protocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs) in the network are required by some components external to the network domain. Usually this information is directly maintained by the ingress Label Edge Routers (LERs) of the MPLS TE LSPs.

One example of using the LSP information is stateful Path Computation Element (PCE) [I-D.ietf-pce-stateful-pce], which could provide benefits in path reoptimization. While some extensions are proposed in Path Computation Element Communication Protocol (PCEP) for the Path Computation Clients (PCCs) to report the LSP states to the PCE, this mechanism may not be applicable in a management-based PCE architecture as specified in section 5.5 of [RFC4655]. As illustrated in the figure below, the PCC is not an LSR in the routing domain, thus the head-end nodes of the TE-LSP may not implement the PCEP protocol. In this case some general mechanism to collect the TE-LSP states from the ingress LERs is needed. This document proposes an LSP state collection mechanism complementary to the mechanism defined in [I-D.ietf-pce-stateful-pce].

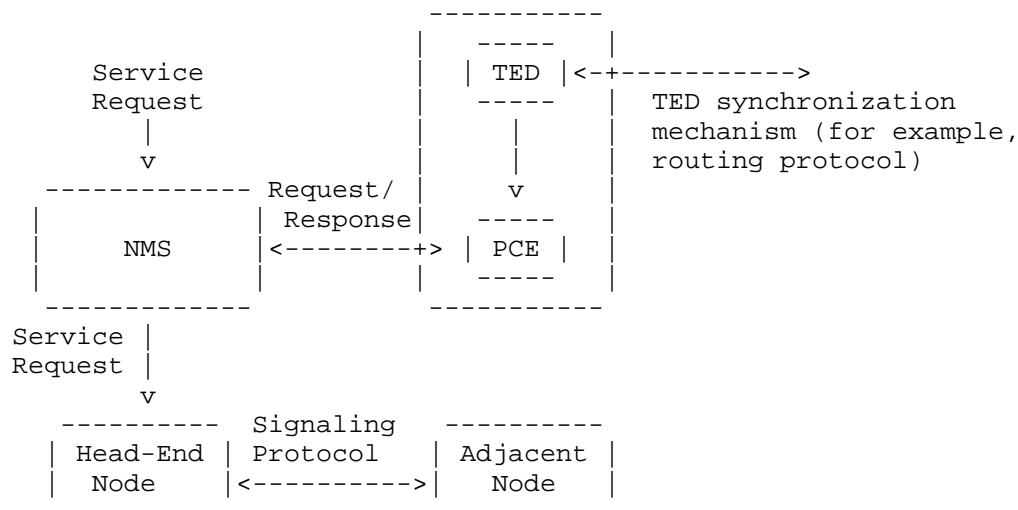


Figure 1. Management-Based PCE Usage

In networks with composite PCE nodes as specified in section 5.1 of [RFC4655], the PCE is implemented on some routers in the network, and the PCCs in the network can use the mechanism described in [I-D.ietf-pce-stateful-pce] to report the LSP information to the PCE

nodes. An external component may further need to collect the LSP information from all the PCEs in the network to get a global view of the LSP states in the network.

In some networks, a centralized controller is used for service placement. Obtaining the TE LSP state information is quite important for making appropriate service placement decisions with the purpose of both meeting the application's requirements and utilizing the network resource efficiently.

The Network Management System (NMS) may need to provide global visibility of the TE LSPs in the network as part of the network visualization.

BGP has been extended to distribute link-state and traffic engineering information and share with some external components [I-D.ietf-idr-ls-distribution]. Using the same protocol to collect other network layer information would be desired by the external components, which avoids introducing multiple protocols for network information collection. This document describes a mechanism to distribute the TE LSP information to external components using BGP.

2. Carrying LSP State Information in BGP

2.1. LSP Information NLRI

A new NLRI "LSP Information NLRI" is advertised in BGP UPDATE messages using the MP_REACH_NLRI and MP_UNREACH_NLRI attributes [RFC4760]. The AFI value is TBD, the SAFI value can be 1 for LSPs in the public network. BGP speakers that wish to exchange LSP Information NLRI MUST use the BGP Multiprotocol Extensions Capability Code (1) to advertise the corresponding (AFI, SAFI) pair, as specified in [RFC4760].

The format of the LSP Information NLRI is as follows:

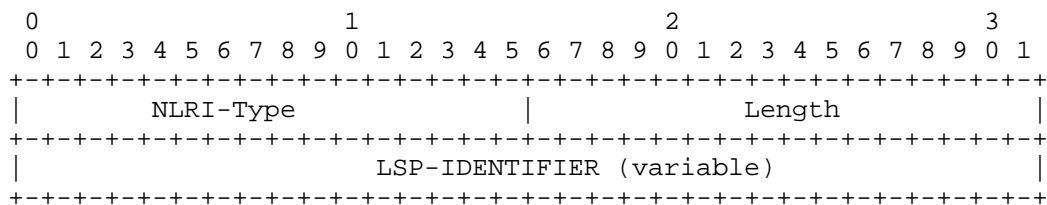


Figure 2. LSP Information NLRI

The NLRI-Type field can be one of the following values:

- o NLRI-Type = 1: IPv4 LSP NLRI
- o NLRI-Type = 2: IPv6 LSP NLRI

If the NLRI-Type value is set to 1, the LSP-IDENTIFIER is the IPv4-LSP-IDENTIFIER structured as below:

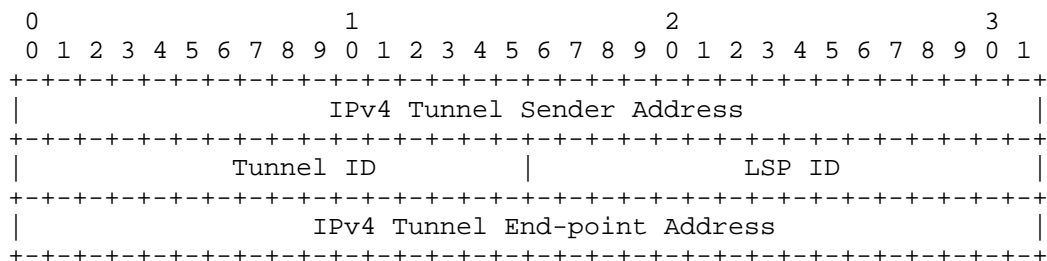


Figure 3. IPv4-LSP-IDENTIFIER

If the NLRI-Type value is set to 2, the LSP-IDENTIFIER is the IPv6-LSP-IDENTIFIER structured as below:

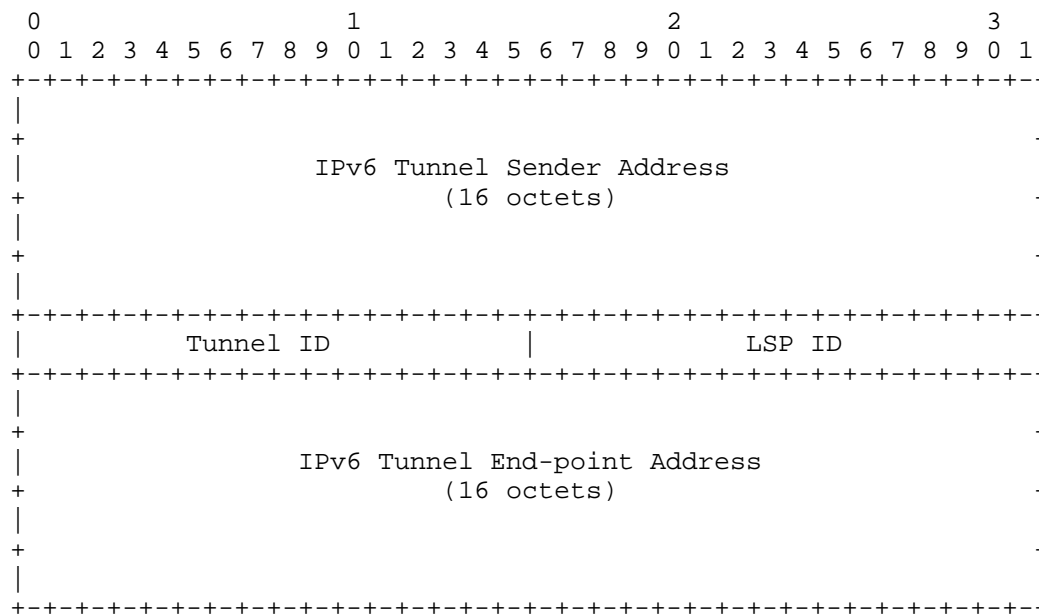


Figure 4. IPv6-LSP-IDENTIFIER

The fields in the IPv4-LSP-IDENTIFIER and IPv6-LSP-IDENTIFIER are the same as specified in [RFC3209].

2.2. LSP State Attribute

The LSP State Attribute is an optional non-transitive BGP attribute which is used to describe the characteristics of the LSPs. The LSP State Attribute consists of a set of objects defined in [RFC3209], [RFC3473] and [RFC5440] . This Attribute SHOULD only be used with the LSP Information NLRI.

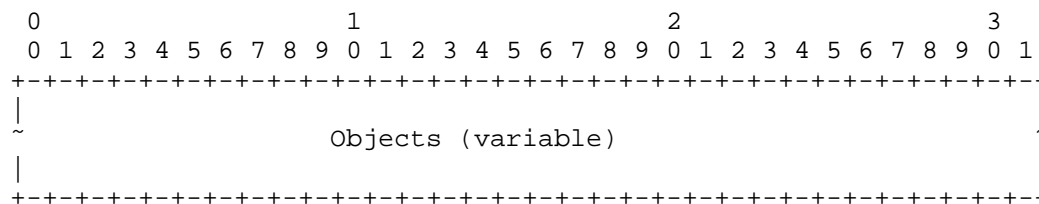


Figure 5. LSP State Attribute

Currently the Objects that can be carried in the LSP State Attribute include:

- o LSP Attributes (LSPA) Object
- o Explicit Route Object (ERO)
- o Record Route Object (RRO)
- o BANDWIDTH Object
- o METRIC Object
- o Protection Object
- o Admin Status Object

Other Objects may also be carried in the LSP State Attribute, which would be specified in a future version.

3. IANA Considerations

IANA needs to assign a new AFI value for the LSP Information NLRI. This code point will come from the "Address Family Numbers" registry.

IANA needs to assign an new code point for the LSP State Attribute from the "BGP Path Attributes" registry.

4. Security Considerations

TBD

5. Acknowledgements

6. References

6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC3473] Berger, L., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic

Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter,
"Multiprotocol Extensions for BGP-4", RFC 4760,
January 2007.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
(PCE) Communication Protocol (PCEP)", RFC 5440,
March 2009.

6.2. Informative References

[I-D.ietf-idr-ls-distribution]
Gredler, H., Medved, J., Previdi, S., and A. Farrel,
"North-Bound Distribution of Link-State and TE Information
using BGP", draft-ietf-idr-ls-distribution-00 (work in
progress), September 2012.

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Medved, J., Varga, R., and I. Minei, "PCEP
Extensions for Stateful PCE",
draft-ietf-pce-stateful-pce-01 (work in progress),
July 2012.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
Element (PCE)-Based Architecture", RFC 4655, August 2006.

Authors' Addresses

Jie Dong
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: jie.dong@huawei.com

Mach(Guoyi) Chen
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: mach.chen@huawei.com

Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2013

H. Gredler
Juniper Networks, Inc.
J. Medved
S. Previdi
Cisco Systems, Inc.
A. Farrel
Juniper Networks, Inc.
S. Ray
Cisco Systems, Inc.
October 22, 2012

North-Bound Distribution of Link-State and TE Information using BGP
draft-ietf-idr-ls-distribution-01

Abstract

In a number of environments, a component external to a network is called upon to perform computations based on the network topology and current state of the connections within the network, including traffic engineering information. This is information typically distributed by IGP routing protocols within the network

This document describes a mechanism by which links state and traffic engineering information can be collected from networks and shared with external components using the BGP routing protocol. This is achieved using a new BGP Network Layer Reachability Information (NLRI) encoding format. The mechanism is applicable to physical and virtual links. The mechanism described is subject to policy control.

Applications of this technique include Application Layer Traffic Optimization (ALTO) servers, and Path Computation Elements (PCEs).

Requirements Language

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

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <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, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

| | |
|-------------------------------------------------------------------------------|----|
| 1. Introduction | 4 |
| 2. Motivation and Applicability | 5 |
| 2.1. MPLS-TE with PCE | 5 |
| 2.2. ALTO Server Network API | 7 |
| 3. Carrying Link State Information in BGP | 8 |
| 3.1. TLV Format | 8 |
| 3.2. The Link State NLRI | 9 |
| 3.2.1. Node Descriptors | 11 |
| 3.2.2. Link Descriptors | 15 |
| 3.2.3. The Prefix NLRI | 16 |
| 3.3. The LINK_STATE Attribute | 16 |
| 3.3.1. Link Attribute TLVs | 16 |
| 3.3.2. Node Attribute TLVs | 20 |
| 3.3.3. Prefix Attributes TLVs | 23 |
| 3.4. BGP Next Hop Information | 27 |
| 3.5. Inter-AS Links | 27 |
| 4. Link to Path Aggregation | 27 |
| 4.1. Example: No Link Aggregation | 27 |
| 4.2. Example: ASBR to ASBR Path Aggregation | 28 |
| 4.3. Example: Multi-AS Path Aggregation | 28 |
| 5. IANA Considerations | 29 |
| 6. Manageability Considerations | 29 |
| 6.1. Operational Considerations | 29 |
| 6.1.1. Operations | 29 |
| 6.1.2. Installation and Initial Setup | 30 |
| 6.1.3. Migration Path | 30 |
| 6.1.4. Requirements on Other Protocols and Functional Components | 30 |
| 6.1.5. Impact on Network Operation | 30 |
| 6.1.6. Verifying Correct Operation | 30 |
| 6.2. Management Considerations | 31 |
| 6.2.1. Management Information | 31 |
| 6.2.2. Fault Management | 31 |
| 6.2.3. Configuration Management | 31 |
| 6.2.4. Accounting Management | 31 |
| 6.2.5. Performance Management | 31 |
| 6.2.6. Security Management | 32 |
| 7. Security Considerations | 32 |
| 8. Acknowledgements | 32 |
| 9. References | 32 |
| 9.1. Normative References | 32 |
| 9.2. Informative References | 34 |
| Authors' Addresses | 34 |

1. Introduction

The contents of a Link State Database (LSDB) or a Traffic Engineering Database (TED) has the scope of an IGP area. Some applications, such as end-to-end Traffic Engineering (TE), would benefit from visibility outside one area or Autonomous System (AS) in order to make better decisions.

The IETF has defined the Path Computation Element (PCE) [RFC4655] as a mechanism for achieving the computation of end-to-end TE paths that cross the visibility of more than one TED or which require CPU-intensive or coordinated computations. The IETF has also defined the ALTO Server [RFC5693] as an entity that generates an abstracted network topology and provides it to network-aware applications.

Both a PCE and an ALTO Server need to gather information about the topologies and capabilities of the network in order to be able to fulfill their function

This document describes a mechanism by which Link State and TE information can be collected from networks and shared with external components using the BGP routing protocol [RFC4271]. This is achieved using a new BGP Network Layer Reachability Information (NLRI) encoding format. The mechanism is applicable to physical and virtual links. The mechanism described is subject to policy control.

A router maintains one or more databases for storing link-state information about nodes and links in any given area. Link attributes stored in these databases include: local/remote IP addresses, local/remote interface identifiers, link metric and TE metric, link bandwidth, reservable bandwidth, per CoS class reservation state, preemption and Shared Risk Link Groups (SRLG). The router's BGP process can retrieve topology from these LSDBs and distribute it to a consumer, either directly or via a peer BGP Speaker (typically a dedicated Route Reflector), using the encoding specified in this document.

The collection of Link State and TE link state information and its distribution to consumers is shown in the following figure.

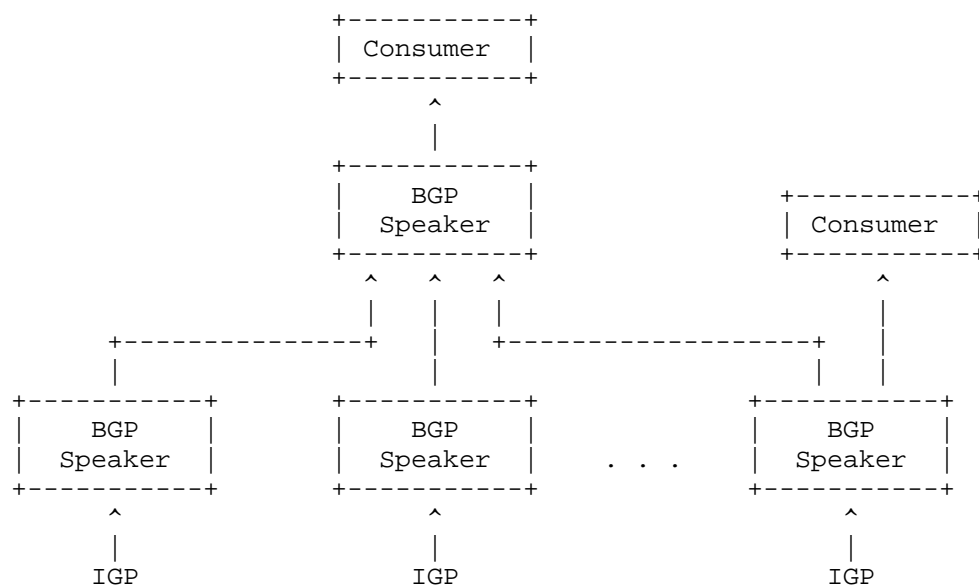


Figure 1: TE Link State info collection

A BGP Speaker may apply configurable policy to the information that it distributes. Thus, it may distribute the real physical topology from the LSDB or the TED. Alternatively, it may create an abstracted topology, where virtual, aggregated nodes are connected by virtual paths. Aggregated nodes can be created, for example, out of multiple routers in a POP. Abstracted topology can also be a mix of physical and virtual nodes and physical and virtual links. Furthermore, the BGP Speaker can apply policy to determine when information is updated to the consumer so that there is reduction of information flow from the network to the consumers. Mechanisms through which topologies can be aggregated or virtualized are outside the scope of this document

2. Motivation and Applicability

This section describes use cases from which the requirements can be derived.

2.1. MPLS-TE with PCE

As described in [RFC4655] a PCE can be used to compute MPLS-TE paths within a "domain" (such as an IGP area) or across multiple domains (such as a multi-area AS, or multiple ASes).

- o Within a single area, the PCE offers enhanced computational power that may not be available on individual routers, sophisticated policy control and algorithms, and coordination of computation across the whole area.
- o If a router wants to compute a MPLS-TE path across IGP areas its own TED lacks visibility of the complete topology. That means that the router cannot determine the end-to-end path, and cannot even select the right exit router (Area Border Router - ABR) for an optimal path. This is an issue for large-scale networks that need to segment their core networks into distinct areas, but which still want to take advantage of MPLS-TE.

Previous solutions used per-domain path computation [RFC5152]. The source router could only compute the path for the first area because the router only has full topological visibility for the first area along the path, but not for subsequent areas. Per-domain path computation uses a technique called "loose-hop-expansion" [RFC3209], and selects the exit ABR and other ABRs or AS Border Routers (ASBRs) using the IGP computed shortest path topology for the remainder of the path. This may lead to sub-optimal paths, makes alternate/back-up path computation hard, and might result in no TE path being found when one really does exist.

The PCE presents a computation server that may have visibility into more than one IGP area or AS, or may cooperate with other PCEs to perform distributed path computation. The PCE obviously needs access to the TED for the area(s) it serves, but [RFC4655] does not describe how this is achieved. Many implementations make the PCE a passive participant in the IGP so that it can learn the latest state of the network, but this may be sub-optimal when the network is subject to a high degree of churn, or when the PCE is responsible for multiple areas.

The following figure shows how a PCE can get its TED information using the mechanism described in this document.

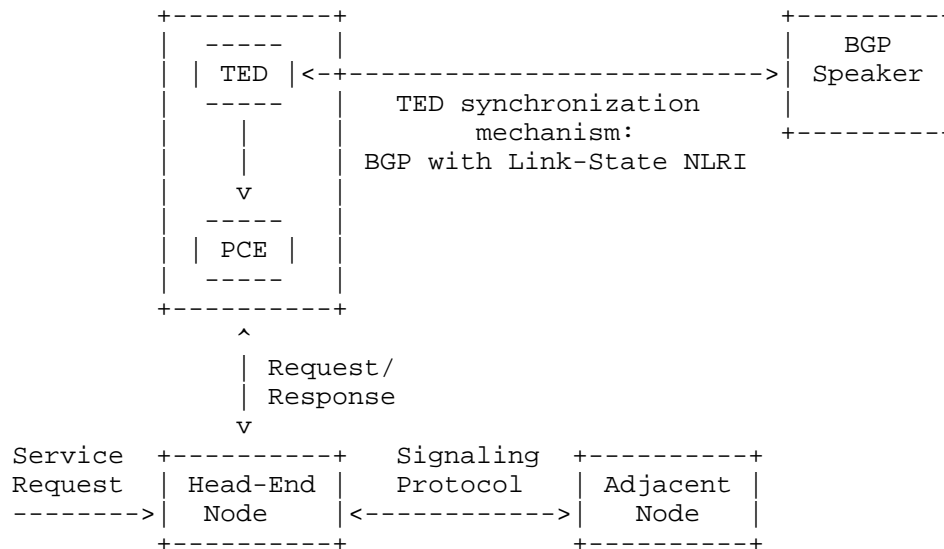


Figure 2: External PCE node using a TED synchronization mechanism

The mechanism in this document allows the necessary TED information to be collected from the IGP within the network, filtered according to configurable policy, and distributed to the PCE as necessary.

2.2. ALTO Server Network API

An ALTO Server [RFC5693] is an entity that generates an abstracted network topology and provides it to network-aware applications over a web service based API. Example applications are p2p clients or trackers, or CDNs. The abstracted network topology comes in the form of two maps: a Network Map that specifies allocation of prefixes to PIDs, and a Cost Map that specifies the cost between PIDs listed in the Network Map. For more details, see [I-D.ietf-alto-protocol].

ALTO abstract network topologies can be auto-generated from the physical topology of the underlying network. The generation would typically be based on policies and rules set by the operator. Both prefix and TE data are required: prefix data is required to generate ALTO Network Maps, TE (topology) data is required to generate ALTO Cost Maps. Prefix data is carried and originated in BGP, TE data is originated and carried in an IGP. The mechanism defined in this document provides a single interface through which an ALTO Server can retrieve all the necessary prefix and network topology data from the underlying network. Note an ALTO Server can use other mechanisms to get network data, for example, peering with multiple IGP and BGP Speakers.

The following figure shows how an ALTO Server can get network topology information from the underlying network using the mechanism described in this document.

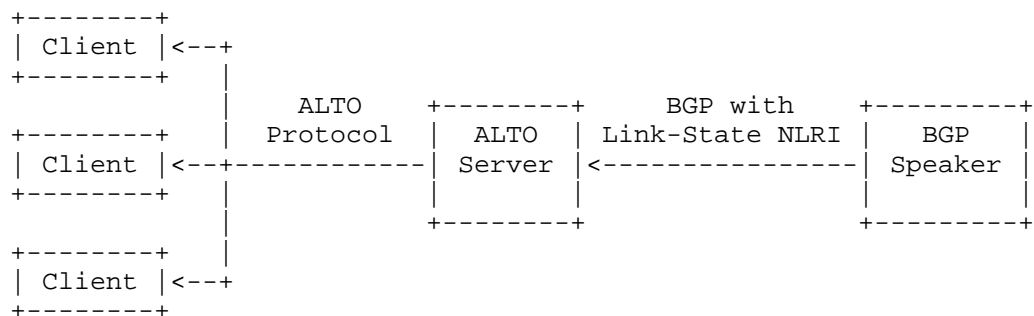


Figure 3: ALTO Server using network topology information

3. Carrying Link State Information in BGP

Two parts: a new BGP NLRI that describes links and nodes comprising IGP link state information, and a new BGP path attribute that carries link and node properties and attributes, such as the link metric or node properties.

3.1. TLV Format

Information in the new link state NLRIs and attributes is encoded in Type/Length/Value triplets. The TLV format is shown in Figure 4.

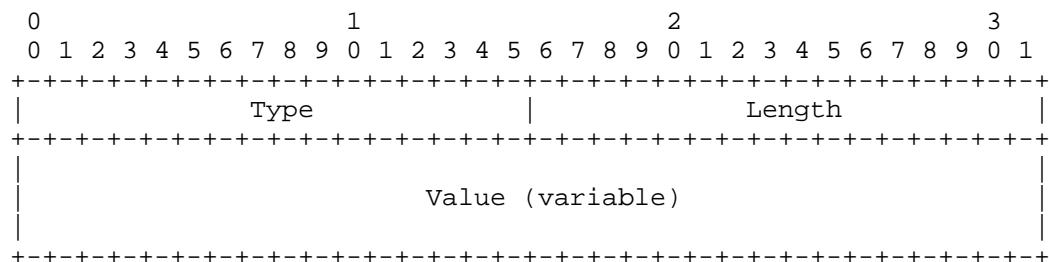


Figure 4: TLV format

The Length field defines the length of the value portion in octets (thus a TLV with no value portion would have a length of zero). The TLV is not padded to four-octet alignment; Unrecognized types are ignored.

3.2. The Link State NLRI

The MP_REACH and MP_UNREACH attributes are BGP's containers for carrying opaque information. Each Link State NLRI describes either a single node or link.

All link, node and prefix information SHALL be encoded using a TBD AFI / TBD SAFI header into those attributes.

In order for two BGP speakers to exchange Link-State NLRI, they MUST use BGP Capabilities Advertisement to ensure that they both are capable of properly processing such NLRI. This is done as specified in [RFC4760], by using capability code 1 (multi-protocol BGP), with an AFI/SAFI TBD.

The format of the Link State NLRI is shown in the following figure.

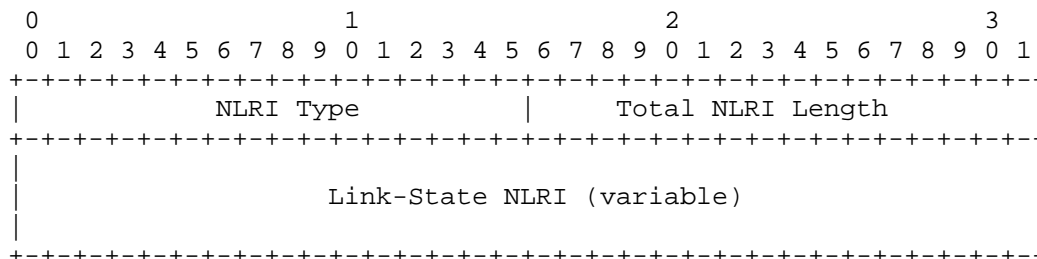


Figure 5: Link State SAFI 1 NLRI Format

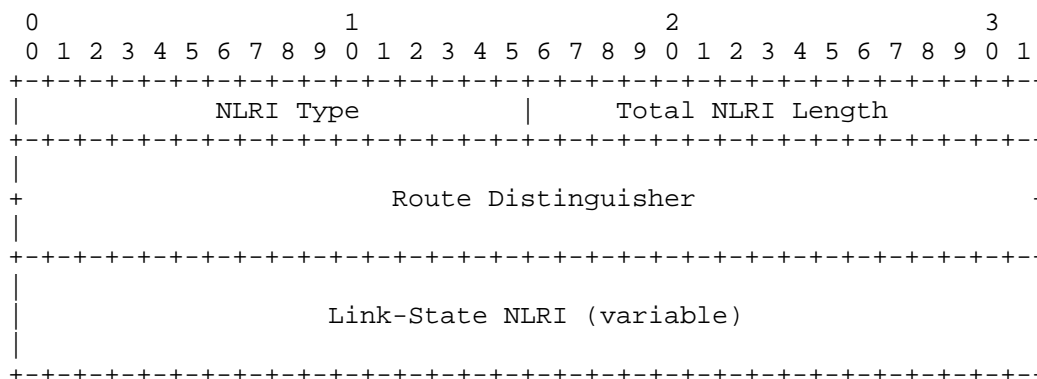


Figure 6: Link State SAFI 128 NLRI Format

The 'Total NLRI Length' field contains the cumulative length of all the TLVs in the NLRI. For VPN applications it also includes the

length of the Route Distinguisher.

The 'NLRI Type' field can contain one of the following values:

Type = 1: Link NLRI, contains link descriptors and link attributes

Type = 2: Node NLRI, contains node attributes

Type = 3: IPv4 Topology Prefix NLRI

Type = 4: IPv6 Topology Prefix NLRI

The Link NLRI (NLRI Type = 1) is shown in the following figure.

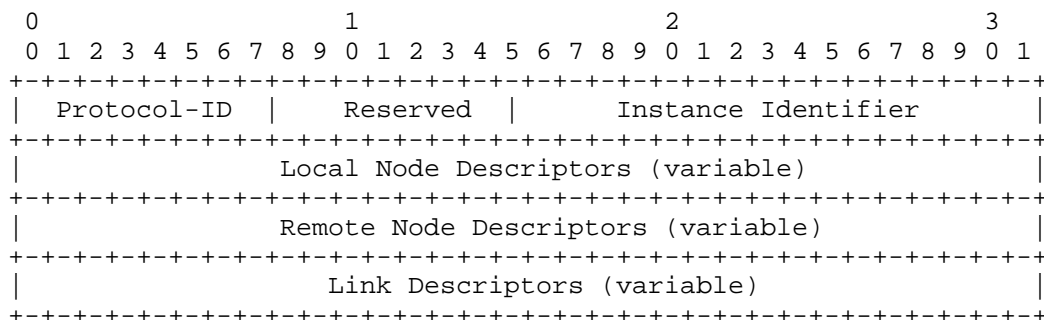


Figure 7: The Link NLRI format

The Node NLRI (NLRI Type = 2) is shown in the following figure.

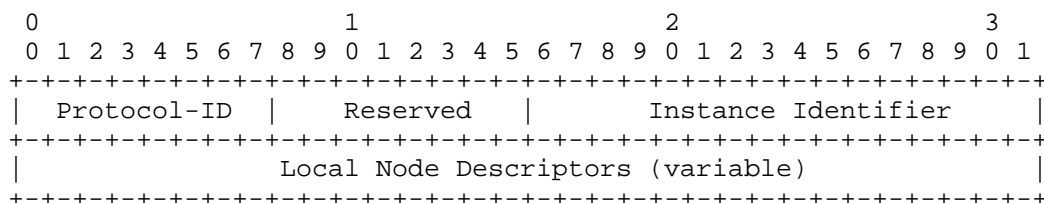


Figure 8: The Node NLRI format

The IPv4 and IPv6 Prefix NLRIs (NLRI Type = 3 and Type = 4) use the same format as shown in the following figure.

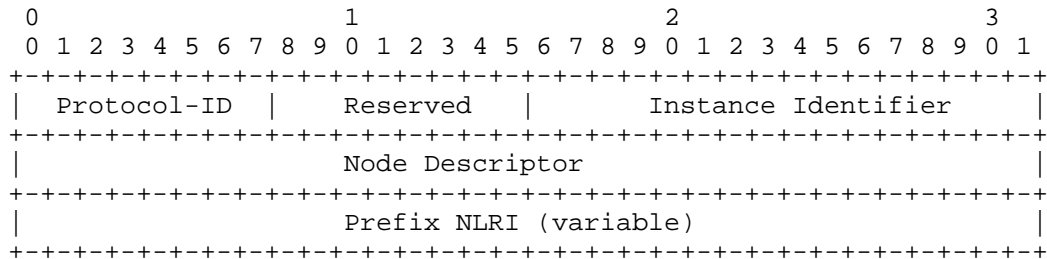


Figure 9: The IPv4/IPv6 Topology Prefix NLRI format

The 'Protocol-ID' field can contain one of the following values:

Protocol-ID = 0: Unknown, The source of NLRI information could not be determined

Protocol-ID: IS-IS Level 1, The NLRI information has been sourced by IS-IS Level 1

Protocol-ID: IS-IS Level 2, The NLRI information has been sourced by IS-IS Level 2

Protocol-ID = 3: OSPF, The NLRI information has been sourced by OSPF

Protocol-ID = 4: Direct, The NLRI information has been sourced from local interface state

Protocol-ID = 5: Static, The NLRI information has been sourced by static configuration

Both OSPF and IS-IS may run multiple routing protocol instances over the same link. See [I-D.ietf-isis-mil] and [RFC6549]. The 'Instance Identifier' field identifies the protocol instance.

Each Node Descriptor and Link Descriptor consists of one or more TLVs described in the following sections. The sender of an UPDATE message MUST order the TLVs within a Node Descriptor or a Link Descriptor in ascending order of TLV type."

3.2.1. Node Descriptors

Each link gets anchored by at least a pair of router-IDs. Since there are many Router-IDs formats (32 Bit IPv4 router-ID, 56 Bit ISO Node-ID and 128 Bit IPv6 router-ID) a link may be anchored by more than one Router-ID pair. The set of Local and Remote Node Descriptors describe which Protocols Router-IDs will be following to

"anchor" the link described by the "Link attribute TLVs". There must be at least one "like" router-ID pair of a Local Node Descriptors and a Remote Node Descriptors per-protocol. If a peer sends an illegal combination in this respect, then this is handled as an NLRI error, described in [RFC4760].

It is desirable that the Router-ID assignments inside the Node anchor are globally unique. However there may be router-ID spaces (e.g. ISO) where not even a global registry exists, or worse, Router-IDs have been allocated following private-IP RFC 1918 [RFC1918] allocation. In order to disambiguate the Router-IDs the local and remote Autonomous System number TLVs of the anchor nodes MUST be included in the NLRI. If the anchor node's AS is a member of an AS Confederation ([RFC5065]), then the Autonomous System number TLV contains the confederations' AS Confederation Identifier and the Member-AS TLV is included in the NLRI. The Local and Remote Autonomous System TLVs are 4 octets wide as described in [RFC4893]. 2-octet AS Numbers SHALL be expanded to 4-octet AS Numbers by zeroing the two MSB octets.

3.2.1.1. Local Node Descriptors

The Local Node Descriptors TLV (Type 256) contains Node Descriptors for the node anchoring the local end of the link. The length of this TLV is variable. The value contains one or more Node Descriptor Sub-TLVs defined in Section 3.2.1.3.

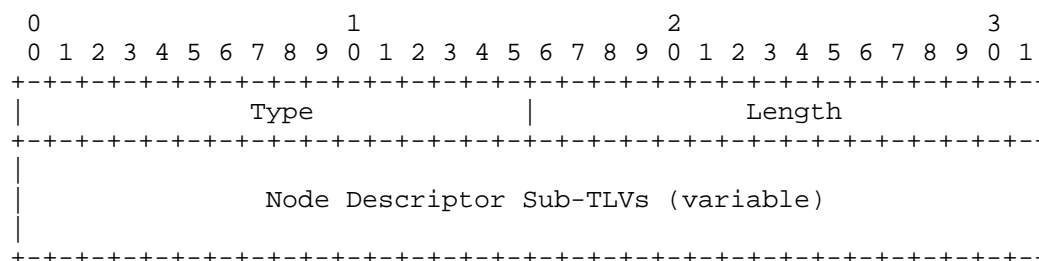


Figure 10: Local Node Descriptors TLV format

3.2.1.2. Remote Node Descriptors

The Remote Node Descriptors TLV (Type 257) contains Node Descriptors for the node anchoring the remote end of the link. The length of this TLV is variable. The value contains one or more Node Descriptor Sub-TLVs defined in Section 3.2.1.3.

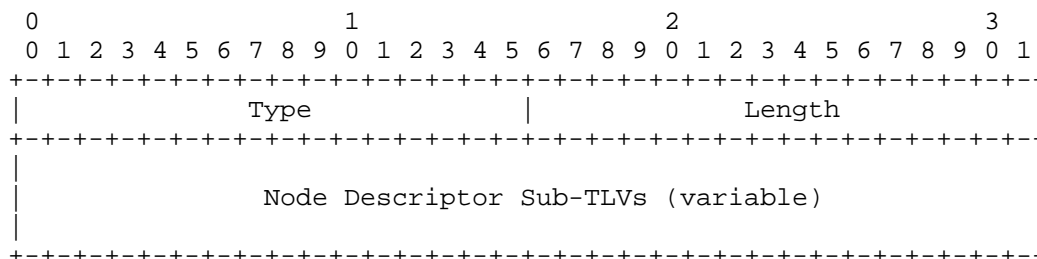


Figure 11: Remote Node Descriptors TLV format

3.2.1.3. Node Descriptor Sub-TLVs

The Node Descriptor Sub-TLV type codepoints and lengths are listed in the following table:

| Type | Description | Length |
|------|-------------------|--------|
| 258 | Autonomous System | 4 |
| 259 | Member-AS | 4 |
| 260 | ISO Node-ID | 7 |
| 261 | IPv4 Router-ID | 5 |
| 262 | IPv4 Router-ID | 17 |

Table 1: Node Descriptor Sub-TLVs

The TLV values in Node Descriptor Sub-TLVs are defined as follows:

Autonomous System: opaque value (32 Bit AS ID)

Member-AS: opaque value (32 Bit AS ID); only included if the node is in an AS confederation.

IPv4 Router ID: opaque value (can be an IPv4 address or an 32 Bit router ID).

IPv6 Router ID: opaque value (can be an IPv6 address or 128 Bit router ID).

ISO Node ID: ISO node-ID (6 octets ISO system-ID) followed by a PSN octet in case LAN "Pseudonode" information gets advertised. The PSN octet must be zero for non-LAN "Pseudonodes".

There can be at most one instance of each TLV type present in any Node Descriptor. The TLV ordering within a Node descriptor MUST be kept in order of increasing numeric value of type. TLVs 258 and 259 specify administrative context in which TLVs 260-262 are to be evaluated. The first TLV from range 260-262 is to be interpreted as the primary node identifier, e.g. it acts as the unique key by which the node can be referenced within its administrative contexts. Any further TLVs are to be treated as secondary identifiers, which may be used for cross-reference, but are to be treated as if they are object attributes.

3.2.1.4. Router-ID Anchoring Example: ISO Pseudonode

IS-IS Pseudonodes are a good example for the variable Router-ID anchoring. Consider Figure 12. This represents a Broadcast LAN between a pair of routers. The "real" (=non pseudonode) routers have both an IPv4 Router-ID and IS-IS Node-ID. The pseudonode does not have an IPv4 Router-ID. Two unidirectional links (Node1, Pseudonode 1) and (Pseudonode 1, Node 2) are being generated.

The NRLI for (Node1, Pseudonode1) encodes local IPv4 router-ID, local ISO node-ID and remote ISO node-id)

The NLRI for (Pseudonode1, Node2) encodes a local ISO node-ID and remote ISO node-id.

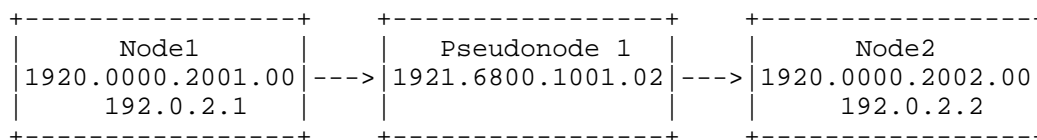


Figure 12: IS-IS Pseudonodes

3.2.1.5. Router-ID Anchoring Example: OSPFv2 to IS-IS Migration

Migrating gracefully from one IGP to another requires congruent operation of both routing protocols during the migration period. The target protocol (IS-IS) supports more router-ID spaces than the source (OSPFv2) protocol. When advertising a point-to-point link between an OSPFv2-only router and an OSPFv2 and IS-IS enabled router the following link information may be generated. Note that the IS-IS router also supports the IPv6 traffic engineering extensions RFC 6119 [RFC6119] for IS-IS.

The NRLI encodes local IPv4 router-id, remote IPv4 router-id, remote ISO node-id and remote IPv6 node-id.

3.2.2. Link Descriptors

The 'Link Descriptor' field is a set of Type/Length/Value (TLV) triplets. The format of each TLV is shown in Section 3.1. The 'Link descriptor' TLVs uniquely identify a link between a pair of anchor Routers. A link described by the Link descriptor TLVs actually is a "half-link", a unidirectional representation of a logical link. In order to fully describe a single logical link two originating routers need to advertise a half-link each, i.e. two link NLRI's will be advertised.

The format and semantics of the 'value' fields in most 'Link Descriptor' TLVs correspond to the format and semantics of value fields in IS-IS Extended IS Reachability sub-TLVs, defined in [RFC5305], [RFC5307] and [RFC6119]. Although the encodings for 'Link Descriptor' TLVs were originally defined for IS-IS, the TLVs can carry data sourced either by IS-IS or OSPF.

The following link descriptor TLVs are valid in the Link NLRI:

| Type | Description | IS-IS TLV/Sub-TLV | Value defined in: |
|------|-------------------------------|----------------------|----------------------|
| 263 | Link Local/Remote Identifiers | 22/4 | [RFC5307]/1.1 |
| 264 | IPv4 interface address | 22/6 | [RFC5305]/3.2 |
| 265 | IPv4 neighbor address | 22/8 | [RFC5305]/3.3 |
| 266 | IPv6 interface address | 22/12 | [RFC6119]/4.2 |
| 267 | IPv6 neighbor address | 22/13 | [RFC6119]/4.3 |
| 268 | Multi Topology ID | --- | Section 3.2.2.1 |

Table 2: Link Descriptor TLVs

3.2.2.1. Multi Topology ID TLV

The Multi Topology ID TLV (Type 268) carries the Multi Topology ID for this link. The semantics of the Multi Topology ID are defined in RFC5120, Section 7.2 [RFC5120], and the OSPF Multi Topology ID), defined in RFC4915, Section 3.7 [RFC4915]. If the value in the Multi Topology ID TLV is derived from OSPF, then the upper 9 bits of the Multi Topology ID are set to 0.

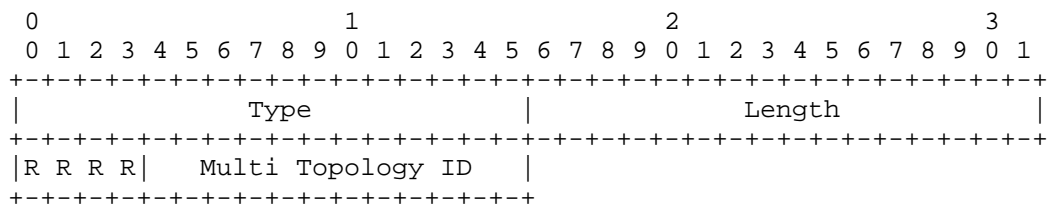


Figure 13: Multi Topology ID TLV format

3.2.3. The Prefix NLRI

The Prefix NLRI is a variable length field that contains an IP address prefix (IPv4 or IPv6) originally advertised in the IGP topology. The distinction between IPv4 and IPv6 prefixes is given by the NLRI Type filed in the Link State NLRI. Reachability information is encoded as one or more 2-tuples of the form <length, prefix>, whose fields are described below:

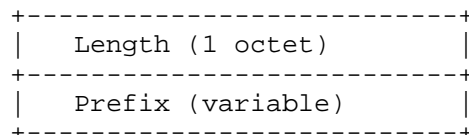


Figure 14: Prefix NLRI format

3.3. The LINK_STATE Attribute

This is an optional, transitive BGP attribute that is used to carry link, node and prefix parameters and attributes. It is defined as a set of Type/Length/Value (TLV) triplets, described in the following section. This attribute SHOULD only be included with Link State NLRIs. This attribute MUST be ignored for all other NLRIs.

3.3.1. Link Attribute TLVs

Each 'Link Attribute' is a Type/Length/Value (TLV) triplet formatted as defined in Section 3.1. The format and semantics of the 'value' fields in some 'Link Attribute' TLVs correspond to the format and semantics of value fields in IS-IS Extended IS Reachability sub-TLVs, defined in [RFC5305] and [RFC5307]. Other 'Link Attribute' TLVs are defined in this document. Although the encodings for 'Link Attribute' TLVs were originally defined for IS-IS, the TLVs can carry data sourced either by IS-IS or OSPF.

The following 'Link Attribute' TLVs are valid in the LINK_STATE attribute:

| Type | Description | IS-IS TLV/Sub-TLV | Defined in: |
|------|-----------------------------------|----------------------|-----------------|
| 269 | Administrative group (color) | 22/3 | [RFC5305]/3.1 |
| 270 | Maximum link bandwidth | 22/9 | [RFC5305]/3.3 |
| 271 | Max. reservable link bandwidth | 22/10 | [RFC5305]/3.5 |
| 272 | Unreserved bandwidth | 22/11 | [RFC5305]/3.6 |
| 273 | Link Protection Type | 22/20 | [RFC5307]/1.2 |
| 274 | MPLS Protocol Mask | --- | Section 3.3.1.1 |
| 275 | Metric | --- | Section 3.3.1.2 |
| 276 | Shared Risk Link Group | --- | Section 3.3.1.3 |
| 277 | OSPF specific link attribute | --- | Section 3.3.1.4 |
| 278 | IS-IS Specific Link Attribute | --- | Section 3.3.1.5 |
| 279 | Area ID | --- | Section 3.3.1.6 |

Table 3: Link Attribute TLVs

3.3.1.1. MPLS Protocol Mask TLV

The MPLS Protocol TLV (Type 274) carries a bit mask describing which MPLS signaling protocols are enabled. The length of this TLV is 1. The value is a bit array of 8 flags, where each bit represents an MPLS Protocol capability.

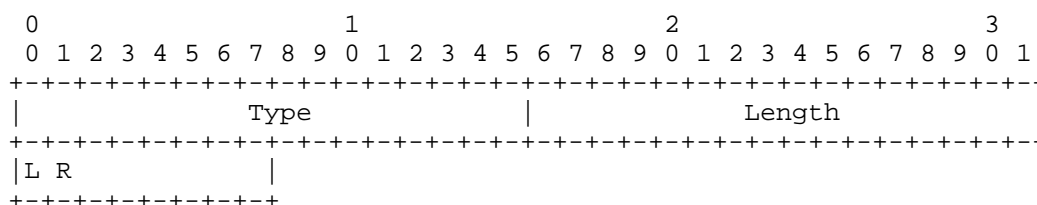


Figure 15: MPLS Protocol TLV

The following bits are defined:

| Bit | Description | Reference |
|-----|---------------------------------------------|-----------|
| 0 | Label Distribution Protocol (LDP) | [RFC5036] |
| 1 | Extension to RSVP for LSP Tunnels (RSVP-TE) | [RFC3209] |
| 2-7 | Reserved for future use | |

Table 4: MPLS Protocol Mask TLV Codes

3.3.1.2. Metric TLV

The IGP Metric TLV (Type 275) carries the metric for this link. The length of this TLV is 3. If the length of the metric from which the IGP Metric value is derived is less than 3 (e.g. for OSPF link metrics or non-wide IS-IS metric), then the upper bits of the TLV are set to 0.

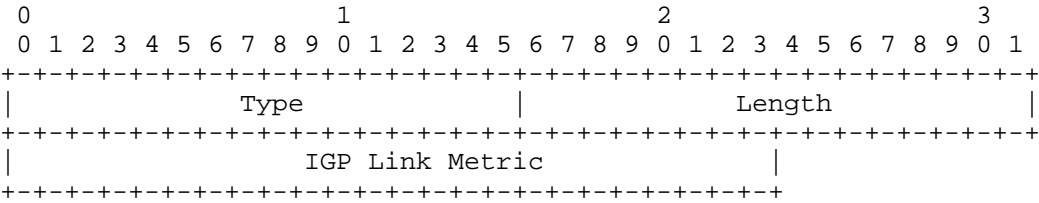


Figure 16: Metric TLV format

3.3.1.3. Shared Risk Link Group TLV

The Shared Risk Link Group (SRLG) TLV (Type 276) carries the Shared Risk Link Group information (see Section 2.3, "Shared Risk Link Group Information", of [RFC4202]). It contains a data structure consisting of a (variable) list of SRLG values, where each element in the list has 4 octets, as shown in Figure 17. The length of this TLV is 4 * (number of SRLG values).

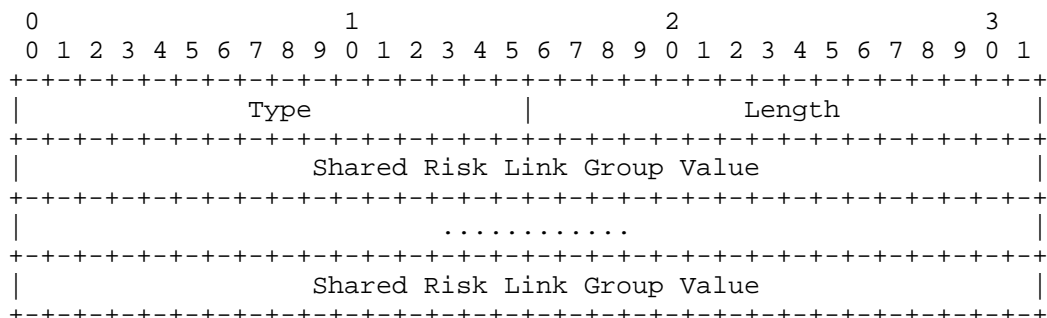


Figure 17: Shared Risk Link Group TLV format

Note that there is no SRLG TLV in OSPF-TE. In IS-IS the SRLG information is carried in two different TLVs: the IPv4 (SRLG) TLV (Type 138) defined in [RFC5307], and the IPv6 SRLG TLV (Type 139) defined in [RFC6119]. Since the Link State NLRI uses variable Router-ID anchoring, both IPv4 and IPv6 SRLG information can be carried in a single TLV.

3.3.1.4. OSPF Specific Link Attribute TLV

The OSPF specific link attribute TLV (Type 277) is an envelope that transparently carries optional link properties TLVs advertised by an OSPF router. The value field contains one or more optional OSPF link attribute TLVs. An originating router shall use this TLV for encoding information specific to the OSPF protocol or new OSPF extensions for which there is no protocol neutral representation in the BGP link-state NLRI.

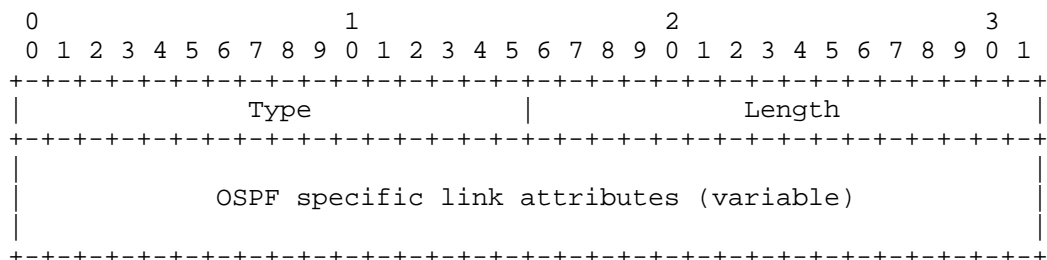


Figure 18: OSPF specific link attribute format

3.3.1.5. IS-IS specific link attribute TLV

The IS-IS specific link attribute TLV (Type 278) is an envelope that transparently carries optional link properties TLVs advertised by an IS-IS router. The value field contains one or more optional IS-IS

link attribute TLVs. An originating router shall use this TLV for encoding information specific to the IS-IS protocol or new IS-IS extensions for which there is no protocol neutral representation in the BGP link-state NLRI.

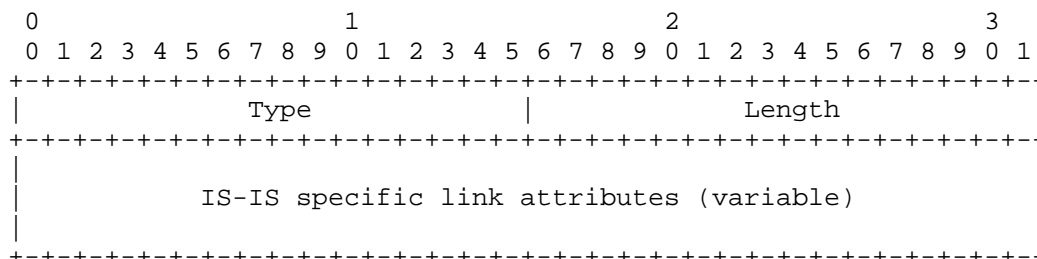


Figure 19: IS-IS specific link attribute format

3.3.1.6. Link Area TLV

The Area TLV (Type 279) carries the Area ID which is assigned on this link. If a link is present in more than one Area then several occurrences of this TLV may be generated. Since only the OSPF protocol carries the notion of link specific areas, the Area ID has a fixed length of 4 octets.

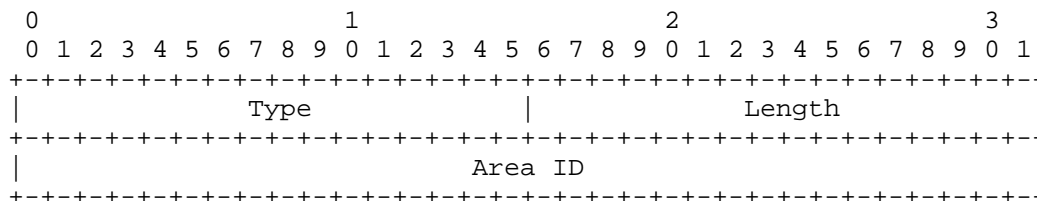


Figure 20: Link Area TLV format

3.3.2. Node Attribute TLVs

The following node attribute TLVs are defined:

| Type | Description | Length |
|------|--------------------------------|----------|
| 280 | Multi Topology | 2 |
| 281 | Node Flag Bits | 1 |
| 282 | OSPF Specific Node Properties | variable |
| 283 | IS-IS Specific Node Properties | variable |
| 284 | Node Area ID | variable |

Table 5: Node Attribute TLVs

3.3.2.1. Multi Topology Node TLV

The Multi Topology TLV (Type 280) carries the Multi Topology ID and topology specific flags for this node. The format and semantics of the 'value' field in the Multi Topology TLV is defined in RFC5120, Section 7.1 [RFC5120]. If the value in the Multi Topology TLV is derived from OSPF, then the upper 9 bits of the Multi Topology ID and the 'O' and 'A' bits are set to 0.

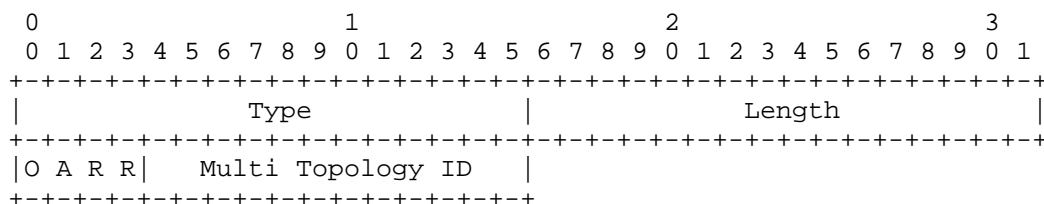


Figure 21: Multi Topology Node TLV format

3.3.2.2. Node Flag Bits TLV

The Node Flag Bits TLV (Type 281) carries a bit mask describing node attributes. The value is a bit array of 8 flags, where each bit represents a node capability.

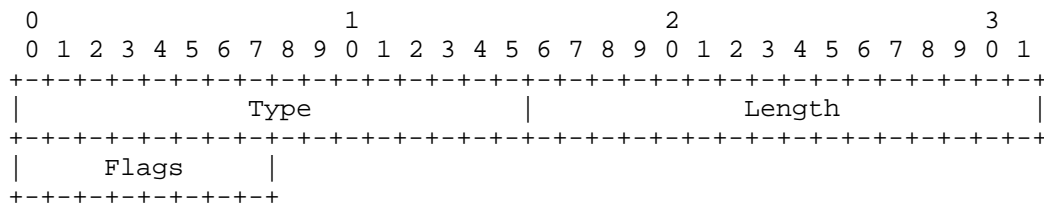


Figure 22: Node Flag Bits TLV format

The bits are defined as follows:

| Bit | Description | Reference |
|-----|--------------|-----------|
| 0 | Overload Bit | [RFC1195] |
| 1 | Attached Bit | [RFC1195] |
| 2 | External Bit | [RFC2328] |
| 3 | ABR Bit | [RFC2328] |

Table 6: Node Flag Bits Definitions

3.3.2.3. OSPF Specific Node Properties TLV

The OSPF Specific Node Properties TLV (Type 282) is an envelope that transparently carries optional node properties TLVs advertised by an OSPF router. The value field contains one or more optional OSPF node property TLVs, such as the OSPF Router Informational Capabilities TLV defined in [RFC4970], or the OSPF TE Node Capability Descriptor TLV described in [RFC5073]. An originating router shall use this TLV for encoding information specific to the OSPF protocol or new OSPF extensions for which there is no protocol neutral representation in the BGP link-state NLRI.

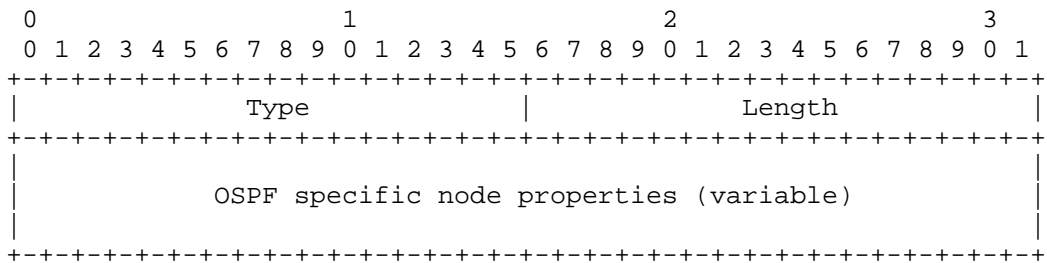


Figure 23: OSPF specific Node property format

3.3.2.4. IS-IS Specific Node Properties TLV

The IS-IS Router Specific Node Properties TLV (Type 283) is an envelope that transparently carries optional node specific TLVs advertised by an IS-IS router. The value field contains one or more optional IS-IS node property TLVs, such as the IS-IS TE Node Capability Descriptor TLV described in [RFC5073]. An originating router shall use this TLV for encoding information specific to the IS-IS protocol or new IS-IS extensions for which there is no protocol neutral representation in the BGP link-state NLRI.

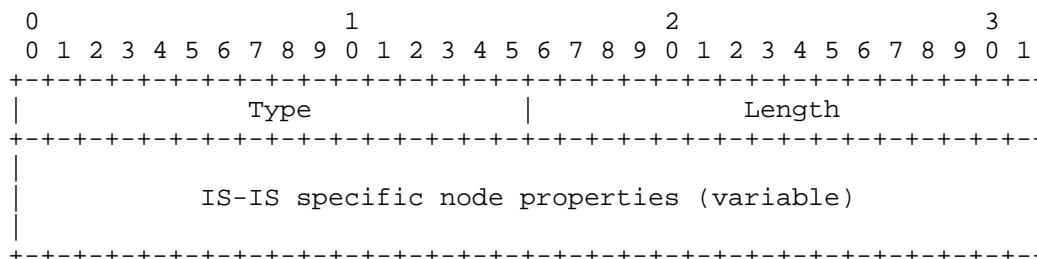


Figure 24: IS-IS specific Node property format

3.3.2.5. Area Node TLV

The Area TLV (Type 284) carries the Area ID which is assigned to this node. If a node is present in more than one Area then several occurrences of this TLV may be generated. Since only the IS-IS protocol carries the notion of per-node areas, the Area ID has a variable length of 1 to 20 octets.

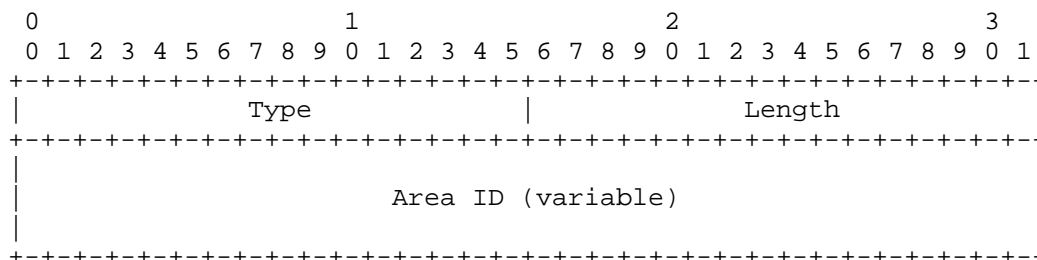


Figure 25: Area Node TLV format

3.3.3. Prefix Attributes TLVs

Prefixes are learned from the IGP topology (ISIS or OSPF) with a set of IGP attributes (such as metric, route tags, route type, etc.) that MUST be reflected into the LINK_STATE attribute. This section describes the different attributes related to the IPv4/IPv6 prefixes. Prefix Attributes TLVs SHOULD be used when advertising NLRI types 3 and 4 only. The following attributes TLVs are defined:

| Type | Description | Length | Reference |
|------|-------------------------|--------|-----------|
| 285 | IGP Flags | 4 | |
| 286 | Route Tag | 4 | [RFC5130] |
| 287 | Extended Tag | 8 | [RFC5130] |
| 288 | Metric | 4 | [RFC5305] |
| 289 | OSPF Forwarding Address | 4 | [RFC2328] |

Table 7: Prefix Attribute TLVs

3.3.3.1. IGP Flags TLV

IGP Flags TLV contains ISIS and OSPF flags and bits originally assigned to the prefix. The IGP Flags TLV is encoded as follows:

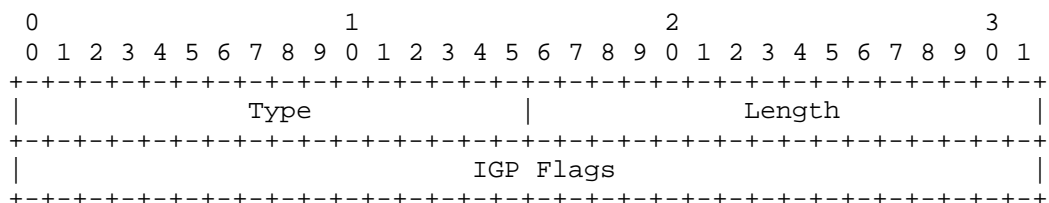


Figure 26: IGP Flag TLV format

where:

Type is 285

Length is 4

The following bits are defined according to the table here below:

| Bit | Description | Reference |
|------|------------------|-----------|
| 0 | ISIS Up/Down Bit | [RFC5305] |
| 1-3 | OSPF Route Type | [RFC2328] |
| 4-15 | RESERVED | |

Table 8: IGP Flag Bits Definitions

OSPF Route Type can be either: Intra-Area (0x1), Inter-Area (0x2), External 1 (0x3), External 2 (0x4), NSSA (0x5) and is encoded in a 3 bits number. For prefixes learned from IS-IS, this field MUST to be

set to 0x0 on transmission.

3.3.3.2. Route Tag

Route Tag TLV carries the original IGP TAG (ISIS or OSPF) of the prefix and is encoded as follows:

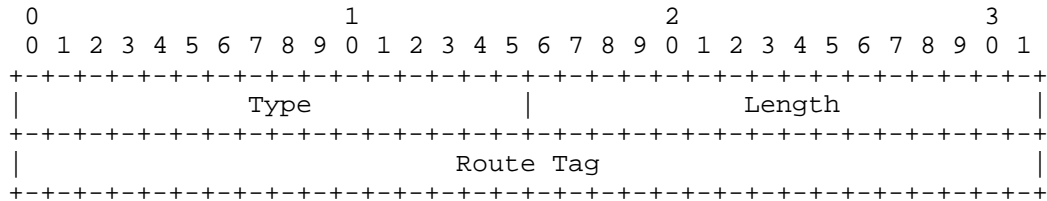


Figure 27: IGP Route TAG TLV format

where:

Type is 286

Length is 4

Route Tag contains the original tags as learned in the IGP topology.

3.3.3.3. Extended Route Tag

Extended Route Tag TLV carries the ISIS Extended Route TAG of the prefix and is encoded as follows:

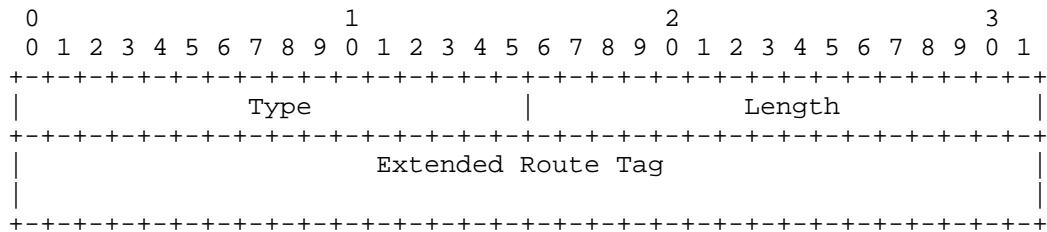


Figure 28: Extended IGP Route TAG TLV format

where:

Type is 287

Length is 8

Extended Route Tag contains the original ISIS Extended Tag as learned

in the IGP topology.

3.3.3.4. Prefix Metric TLV

Prefix Metric TLV carries the metric of the prefix as known in the IGP topology. The attribute is mandatory and can only appear once.

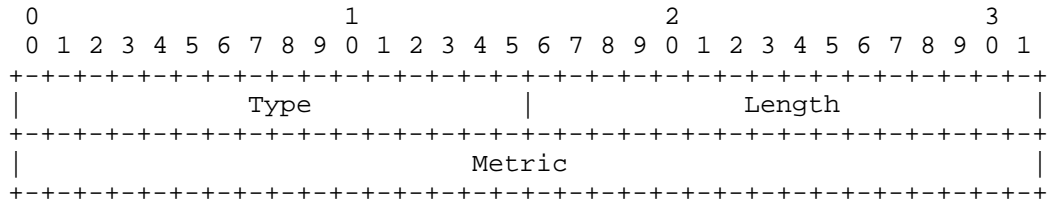


Figure 29: Prefix Metric TLV Format

where:

Type is 288

Length is 4

3.3.3.5. OSPF Forwarding Address TLV

OSPF Forwarding Address TLV carries the OSPF forwarding address as known in the original OSPF advertisement. Forwarding address can be either IPv4 or IPv6.

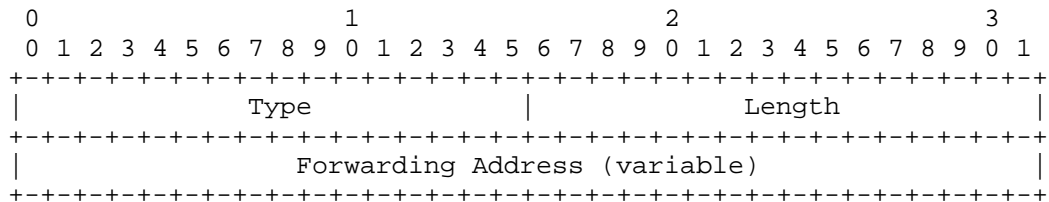


Figure 30: OSPF Forwarding Address TLV Format

where:

Type is 289

Length is 4 for an IPv4 forwarding address an 16 for an IPv6 forwarding address

3.4. BGP Next Hop Information

BGP link-state information for both IPv4 and IPv6 networks can be carried over either an IPv4 BGP session, or an IPv6 BGP session. If IPv4 BGP session is used, then the next hop in the MP_REACH_NLRI SHOULD be an IPv4 address. Similarly, if IPv6 BGP session is used, then the next hop in the MP_REACH_NLRI SHOULD be an IPv6 address. Usually the next hop will be set to the local end-point address of the BGP session. The next hop address MUST be encoded as described in [RFC4760]. The length field of the next hop address will specify the next hop address-family. If the next hop length is 4, then the next hop is an IPv4 address; if the next hop length is 16, then it is a global IPv6 address and if the next hop length is 32, then there is one global IPv6 address followed by a link-local IPv6 address. The link-local IPv6 address should be used as described in [RFC2545].

3.5. Inter-AS Links

The main source of TE information is the IGP, which is not active on inter-AS links. In order to inject a non-IGP enabled link into the BGP link-state RIB an implementation must support configuration of static links.

4. Link to Path Aggregation

Distribution of all links available in the global Internet is certainly possible, however not desirable from a scaling and privacy point of view. Therefore an implementation may support link to path aggregation. Rather than advertising all specific links of a domain, an ASBR may advertise an "aggregate link" between a non-adjacent pair of nodes. The "aggregate link" represents the aggregated set of link properties between a pair of non-adjacent nodes. The actual methods to compute the path properties (of bandwidth, metric) are outside the scope of this document. The decision whether to advertise all specific links or aggregated links is an operator's policy choice. To highlight the varying levels of exposure, the following deployment examples shall be discussed.

4.1. Example: No Link Aggregation

Consider Figure 31. Both AS1 and AS2 operators want to protect their inter-AS {R1,R3}, {R2, R4} links using RSVP-FRR LSPs. If R1 wants to compute its link-protection LSP to R3 it needs to "see" an alternate path to R3. Therefore the AS2 operator exposes its topology. All BGP TE enabled routers in AS1 "see" the full topology of AS and therefore can compute a backup path. Note that the decision if the direct link between {R3, R4} or the {R4, R5, R3} path is used is made

by the computing router.

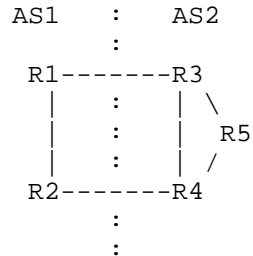


Figure 31: no-link-aggregation

4.2. Example: ASBR to ASBR Path Aggregation

The brief difference between the "no-link aggregation" example and this example is that no specific link gets exposed. Consider Figure 32. The only link which gets advertised by AS2 is an "aggregate" link between R3 and R4. This is enough to tell AS1 that there is a backup path. However the actual links being used are hidden from the topology.

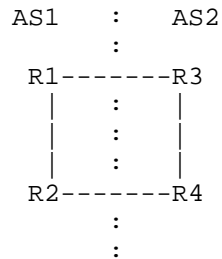


Figure 32: asbr-link-aggregation

4.3. Example: Multi-AS Path Aggregation

Service providers in control of multiple ASes may even decide to not expose their internal inter-AS links. Consider Figure 33. Rather than exposing all specific R3 to R6 links, AS3 is modeled as a single node which connects to the border routers of the aggregated domain.

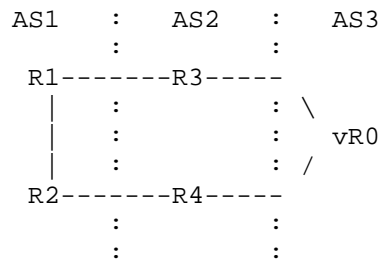


Figure 33: multi-as-aggregation

5. IANA Considerations

This document requests a code point from the registry of Address Family Numbers.

This document requests a code point from the BGP Path Attributes registry.

This document requests creation of a new registry for node anchor, link descriptor and link attribute TLVs. Values 0-255 are reserved. Values 256-65535 will be used for Codepoints. The registry will be initialized as shown in Table 2 and Table 3. Allocations within the registry will require documentation of the proposed use of the allocated value and approval by the Designated Expert assigned by the IESG (see [RFC5226]).

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Manageability Considerations

This section is structured as recommended in [RFC5706].

6.1. Operational Considerations

6.1.1. Operations

Existing BGP operation procedures apply. No new operation procedures are defined in this document. It shall be noted that the NLRI information present in this document purely carries application level data that have no immediate corresponding forwarding state impact. As such, any churn in reachability information has different impact than regular BGP update which needs to change forwarding state for an entire router. Furthermore it is anticipated that distribution of

this NLRI will be handled by dedicated route-reflectors providing a level of isolation and fault-containment between different NLRI types.

6.1.2. Installation and Initial Setup

Configuration parameters defined in Section 6.2.3 SHOULD be initialized to the following default values:

- o The Link-State NLRI capability is turned off for all neighbors.
- o The maximum rate at which Link State NLRIs will be advertised/withdrawn from neighbors is set to 200 updates per second.

6.1.3. Migration Path

The proposed extension is only activated between BGP peers after capability negotiation. Moreover, the extensions can be turned on/off on an individual peer basis (see Section 6.2.3), so the extension can be gradually rolled out in the network.

6.1.4. Requirements on Other Protocols and Functional Components

The protocol extension defined in this document does not put new requirements on other protocols or functional components.

6.1.5. Impact on Network Operation

Frequency of Link-State NLRI updates could interfere with regular BGP prefix distribution. A network operator MAY use a dedicated Route-Reflector infrastructure to distribute Link-State NLRIs.

Distribution of Link-State NLRIs SHOULD be limited to a single admin domain, which can consist of multiple areas within an AS or multiple ASes.

6.1.6. Verifying Correct Operation

Existing BGP procedures apply. In addition, an implementation SHOULD allow an operator to:

- o List neighbors with whom the Speaker is exchanging Link-State NLRIs

6.2. Management Considerations

6.2.1. Management Information

6.2.2. Fault Management

TBD.

6.2.3. Configuration Management

An implementation SHOULD allow the operator to specify neighbors to which Link-State NLRIs will be advertised and from which Link-State NLRIs will be accepted.

An implementation SHOULD allow the operator to specify the maximum rate at which Link State NLRIs will be advertised/withdrawn from neighbors

An implementation SHOULD allow the operator to specify the maximum rate at which Link State NLRIs will be accepted from neighbors

An implementation SHOULD allow the operator to specify the maximum number of Link State NLRIs stored in router's RIB.

An implementation SHOULD allow the operator to create abstracted topologies that are advertised to neighbors; Create different abstractions for different neighbors.

6.2.4. Accounting Management

Not Applicable.

6.2.5. Performance Management

An implementation SHOULD provide the following statistics:

- o Total number of Link-State NLRI updates sent/received
- o Number of Link-State NLRI updates sent/received, per neighbor
- o Number of errored received Link-State NLRI updates, per neighbor
- o Total number of locally originated Link-State NLRIs

6.2.6. Security Management

An operator SHOULD define ACLs to limit inbound updates as follows:

- o Drop all updates from Consumer peers

7. Security Considerations

Procedures and protocol extensions defined in this document do not affect the BGP security model.

A BGP Speaker SHOULD NOT accept updates from a Consumer peer.

An operator SHOULD employ a mechanism to protect a BGP Speaker against DDOS attacks from Consumers.

8. Acknowledgements

We would like to thank Nischal Sheth, Alia Atlas, Robert Varga, David Ward, Derek Yeung, Murtuza Lightwala, John Scudder, Kaliraj Vairavakkalai, Les Ginsberg, Liem Nguyen, Manish Bhardwaj, Mike Shand, Peter Psenak, Rex Fernando, Richard Woundy, Saikat Ray, Steven Luong, Tamas Mondal, Waqas Alam, and Yakov Rekhter for their comments.

9. References

9.1. Normative References

- [RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.
- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.
- [RFC2545] Marques, P. and F. Dupont, "Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing", RFC 2545, March 1999.

- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC4202] Kompella, K. and Y. Rekhter, "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4202, October 2005.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.
- [RFC4893] Vohra, Q. and E. Chen, "BGP Support for Four-octet AS Number Space", RFC 4893, May 2007.
- [RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, June 2007.
- [RFC5036] Andersson, L., Minei, I., and B. Thomas, "LDP Specification", RFC 5036, October 2007.
- [RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, August 2007.
- [RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, February 2008.
- [RFC5130] Previdi, S., Shand, M., and C. Martin, "A Policy Control Mechanism in IS-IS Using Administrative Tags", RFC 5130, February 2008.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.
- [RFC5307] Kompella, K. and Y. Rekhter, "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, October 2008.
- [RFC6119] Harrison, J., Berger, J., and M. Bartlett, "IPv6 Traffic

Engineering in IS-IS", RFC 6119, February 2011.

9.2. Informative References

- [I-D.ietf-alto-protocol] Alimi, R., Penno, R., and Y. Yang, "ALTO Protocol", draft-ietf-alto-protocol-13 (work in progress), September 2012.
- [I-D.ietf-isis-mi] Previdi, S., Ginsberg, L., Shand, M., Roy, A., and D. Ward, "IS-IS Multi-Instance", draft-ietf-isis-mi-08 (work in progress), October 2012.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.
- [RFC4970] Lindem, A., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 4970, July 2007.
- [RFC5073] Vasseur, J. and J. Le Roux, "IGP Routing Protocol Extensions for Discovery of Traffic Engineering Node Capabilities", RFC 5073, December 2007.
- [RFC5152] Vasseur, JP., Ayyangar, A., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, February 2008.
- [RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", RFC 5693, October 2009.
- [RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", RFC 5706, November 2009.
- [RFC6549] Lindem, A., Roy, A., and S. Mirtorabi, "OSPFv2 Multi-Instance Extensions", RFC 6549, March 2012.

Authors' Addresses

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

Email: hannes@juniper.net

Jan Medved
Cisco Systems, Inc.
170, West Tasman Drive
San Jose, CA 95134
US

Email: jmedved@cisco.com

Stefano Previdi
Cisco Systems, Inc.
Via Del Serafico, 200
Rome 00142
Italy

Email: sprevidi@cisco.com

Adrian Farrel
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: afarrel@juniper.net

Saikat Ray
Cisco Systems, Inc.
170, West Tasman Drive
San Jose, CA 95134
US

Email: sairay@cisco.com

Internet Engineering Task Force
Internet-Draft
Updates: 4456 (if approved)
Intended status: Standards Track
Expires: April 21, 2013

J. Scudder
Juniper Networks
October 18, 2012

Considerations for Route Reflection and EBGP
draft-scudder-idr-ebgp-rr-02

Abstract

Although originally conceived of as a purely IBGP device, in some cases a route reflector may function as an EBGP speaker in addition to its role as envisioned in RFC 4456. When it does so, just like any other EBGP speaker it must advertise its routes to its IBGP peers. This document updates RFC 4456 by explaining what behavior is required of a route reflector that also functions as an EBGP speaker. It also clarifies the use of the ORIGINATOR_ID and CLUSTER_LIST attributes in this environment.

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 21, 2013.

Copyright Notice

Copyright (c) 2012 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.

1. Introduction

Although originally conceived of as a purely IBGP device, in some cases a BGP [RFC4271] route reflector may function as an EBG speaker in addition to its role as envisioned in [RFC4456]. When it does so, just like any other EBG speaker it must advertise its routes to its IBGP peers. This document updates RFC 4456 by explaining what behavior is required of a route reflector that also functions as an EBG speaker. It also clarifies the use of the ORIGINATOR_ID and CLUSTER_LIST attributes in this environment.

The cardinal observation is that the functions outlined in [RFC4456] apply exclusively to "reflected" routes, that is, IBGP routes that are propagated to IBGP peers.

1.1. Requirements Language

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

2. Terminology

In addition to the terms defined in [RFC4271] Section 1.1 and in [RFC4456], this document makes use of the following:

ASBR: Autonomous System Border Router. See EBG Speaker.

RR: Route Reflector.

Redundant Route Reflector (or Redundant RR): Another route reflector in the same cluster as the route reflector under consideration, when both route reflectors are configured with the same CLUSTER_ID.

EBG Speaker: A BGP speaker that has one or more EBG peerings, and thereby learns one or more EBG routes. (If no routes are learned it is still an EBG Speaker, but this is a case of "a tree falling in a forest with no one to hear it.") ASBRs are EBG speakers, although not all EBG speakers are ASBRs.

3. Updates to RFC 4456

When deciding how a route reflector that is also an EBG speaker should propagate EBG routes into IBGP, the key observation is that [RFC4456] deals only with "reflected" routes, i.e. IBGP routes that are propagated into IBGP. For EBG-learned routes, the BGP speaker is the only source of routes for its AS. For this reason, the restrictions and assumptions that apply to reflected routes do not apply to EBG-learned routes. For the purposes of such routes, the BGP speaker functions as a normal IBGP router. For example, the [RFC4456] stricture against modifying the NEXT_HOP, AS_PATH, LOCAL_PREF, and MED attributes does not apply to EBG-learned routes that are propagated into IBGP.

Specific updates to [RFC4456] are:

- o The speaker MUST NOT add a CLUSTER_LIST to EBG-learned routes when advertising them into IBGP.
- o The attributes ORIGINATOR_ID and CLUSTER_LIST MUST NOT be sent to EBG peers. If received from an EBG peer, these attributes MUST be ignored and not propagated further; an error message MAY be logged.

4. Deployment Considerations

If route reflectors are deployed in an Autonomous System such that no two route reflectors have the same CLUSTER_ID, then there are no "redundant route reflectors" (as the term is used herein) and thus, the considerations regarding redundant RRs below are moot.

A RR that serves as an EBG speaker SHOULD have an IBGP peering with any redundant RR. It SHOULD advertise the same EBG-learned routes over this peering that it advertises to any other IBGP peer. It MAY suppress reflection of any IBGP-learned routes to the redundant RR. (Recall that according to [RFC4456] Section 8, such routes would be ignored by the redundant RR anyway due to a loop in the CLUSTER_LIST.) The peering MAY be omitted if the route reflectors in question are control plane-only devices not in the forwarding path of any traffic, or if the network in question uses some form of tunneled or label-switched forwarding. The cost of omitting the peering is that certain low-probability failure modes may cause unnecessary unreachability -- specifically, if the EBG-speaking RR were to lose its session to one or more of its RR clients, reachability to the EBG-learned routes would be preserved if a session remained up to its redundant RR peer. (Similar considerations apply even to route reflectors which do not have a collocated EBG speaker function, but

such are beyond the scope of this document.)

5. IANA Considerations

This document makes no request of IANA.

6. Security Considerations

This clarification to BGP does not change the underlying security issues.

7. Acknowledgements

The author would like to thank Serpil Bayraktar, Jeff Haas, Senad Palislaamovic, Yakov Rekhter, Jim Uttaro and Kaliraj Vairavakkalai for their input.

8. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, April 2006.

Author's Address

John Scudder
Juniper Networks
Email: jgs@juniper.net

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

G. Van de Velde

K. Patel
D. Rao
Cisco Systems
R. Raszuk
NTT MCL Inc.
R. Bush
Internet Initiative Japan
March 9, 2015

BGP Remote-Next-Hop
draft-vandavelde-idr-remote-next-hop-09

Abstract

The BGP Remote-Next-Hop attribute is an optional transitive attribute intended to facilitate automatic tunnelling across an AS for an NLRI in a given address family. The attribute carries one or more tunnel end-points and associated tunnel encapsulation information for a NLRI.

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 September 10, 2015.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

| | |
|------------------------------------------------------------------------------------|----|
| 1. Introduction | 3 |
| 2. Requirements Language | 3 |
| 3. Remote-Next-Hop Attribute | 3 |
| 3.1. Tunnel Encapsulation attribute versus BGP Remote-Next-Hop attribute | 4 |
| 4. BGP Remote-Next-Hop attribute TLV Format | 4 |
| 5. Encapsulation sub-TLVs for virtual network overlays | 5 |
| 5.1. Encapsulation sub-TLV for VXLAN | 6 |
| 5.2. Encapsulation sub-TLV for NVGRE | 7 |
| 5.3. Encapsulation sub-TLV for GTP | 8 |
| 5.4. Encapsulation for MPLS-in-GRE | 8 |
| 6. Remote-Next-Hop Bestpath Considerations | 9 |
| 7. Securing Remote-Next-Hop | 9 |
| 7.1. Restrictions on Announcing of Remote-Next-Hop Attribute | 10 |
| 7.2. Restrictions on Originating of Remote-Next-Hop Attribute | 10 |
| 8. Multiple tunnel endpoint addresses | 11 |
| 9. Attribute error handling | 11 |
| 10. BGP speakers that do not support BGP Remote-Next-Hop attribute | 11 |
| 11. Use Case scenarios | 11 |
| 11.1. Stateless user-plane architecture for virtualized EPC (vEPC) | 12 |
| 11.2. Stateless User-plane Architecture for virtual Packet Edge | 12 |
| 11.3. Dynamic Network Overlay Infrastructure | 12 |
| 11.4. Simple VPN solution using Multi-point Security Association | 12 |
| 12. IANA Considerations | 13 |
| 13. Security Considerations | 13 |
| 14. Privacy Considerations | 14 |
| 15. Acknowledgements | 14 |
| 16. Change Log | 14 |
| 17. References | 14 |
| 17.1. Normative References | 14 |
| 17.2. Informative References | 15 |
| Authors' Addresses | 16 |

1. Introduction

[RFC5512] defines an attribute attached to an NLRI to signal tunnel end-point encapsulation information between two BGP speakers for a single tunnel. [RFC5512] requires that a new address-family needs to be enabled between the two BGP speakers. It also assumes that the exchanged tunnel endpoint is the NLRI.

This document defines a new BGP transitive attribute known as a Remote-Next-Hop BGP attribute for Intra-AS and Inter-AS usage, and simplifies the exchange and operations involved with tunnel end-point information propagation between two BGP speakers.

The tunnel endpoint information and the tunnel encapsulation information is carried within a Remote-Next-Hop BGP attribute. This attribute can be added to any BGP NLRI. This way the Address Family (AF) of the NLRI exchanged is decoupled from the tunnel SAFI address-family defined in [RFC5512]. Multiple Remote-Next-Hop attribute TLVs can be added to a single NLRI.

Security measures SHOULD be taken to protect against accidental or malicious tampering of the Remote-Next-Hop attribute.

2. 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 any normative meaning.

3. Remote-Next-Hop Attribute

There are an increasing number of use cases where the exchange of multiple unique tunnel endpoints and associated tunnel data is desired for a prefix using segments of an existing infrastructure, where requiring a new address-family to be enabled would add operational complexity.

The BGP Remote-Next-Hop attribute is defined to be attached to each originated BGP NLRI in any applicable address-family. Multiple Remote-Next-Hop attribute TLVs can be applied to a single originated BGP NLRI. Each TLV can contain one or more sub-TLVs that carry encapsulation information. Thus, it enables a simple mechanism to signal multiple, unique tunnel endpoints for a given prefix; as well as multiple encapsulation parameters for prefixes with the same remote tunnel end-point.

BGP Remote-Next-Hop attribute is a Transitive Optional BGP attribute, allowing to signal next-hop encapsulation parameters in a transitive manner without the requirement to enable a new address-family.

This document specifies the tunnel types that can be used with this attribute. The sub-TLVs from [RFC5512] and BGP IPsec tunnel encapsulation [RFC5566] are reused for the BGP Next-Hop-Attribute.

3.1. Tunnel Encapsulation attribute versus BGP Remote-Next-Hop attribute

The use of Tunnel Encapsulation attribute [RFC5512] is based on the principle that the tunnel end-point is carried as part of BGP NLRI in an Encapsulation SAFI.

This requires enabling of the Encapsulation SAFI within a BGP enabled network. It also sets up an interdependency between BGP routes in different SAFIs and the BGP Tunnel SAFI for resolving tunnel next-hops.

The Encapsulation SAFI [RFC5512] assumes that the tunnel endpoint is the NLRI exchanged in the Encaps SAFI, while Remote-Next-Hop decouples the exchanged NLRI from the tunnel end-point information, thereby requiring mutual exclusive usage of the two mechanisms.

While [RFC5512] allows multiple tunnel endpoints and multiple tunnel types to be carried within a BGP Encaps SAFI, the correlation of Tunnel information with other SAFIs is done using the color extended community which is also non-trivial.

4. BGP Remote-Next-Hop attribute TLV Format

This attribute is an optional transitive attribute [RFC1771].

The BGP Remote-Next-Hop attribute is composed of a set of Type-Length-Value (TLV) encodings. The type code of the attribute is (IANA to assign). Each TLV contains information corresponding to a particular tunnel end-point address.

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Tunnel Type (2 Octets)   |                               Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Addr len   |   Tunnel Address (IPv4 or IPv6)   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     AS Number   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

```

|                               Tunnel Parameters                               |
~-----~
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Tunnel Type (2 octets): identifies the type of tunneling technology being signaled. This document specifies the following types:

- L2TPv3 over IP [RFC3931]: Tunnel Type = 1
- GRE [RFC2784]: Tunnel Type = 2
- Transmit tunnel endpoint [RFC5566]: Tunnel Type = 3
- IPsec in Tunnel-mode [RFC5566]: Tunnel Type = 4
- IP in IP tunnel
 - with IPsec Transport Mode [RFC5566]: Tunnel Type = 5
- MPLS-in-IP tunnel
 - with IPsec Transport Mode [RFC5566]: Tunnel Type = 6
- IP in IP [RFC2003] [RFC4213]: Tunnel Type = 7

This document defines the following types:

- VXLAN: Tunnel Type = 8
- NVGRE: Tunnel Type = 9
- GTP: Tunnel Type = 10
- MPLS-in-GRE: Tunnel Type = 11
- MPLS-in-UDP: Tunnel Type = 12
- MPLS-in-UDP-with-DTLS: Tunnel Type = 13

Unknown types MUST be ignored and skipped upon receipt.

Length (2 octets): the total number of octets of the value field.

Tunnel Address Length (1 octet): Length of Tunnel Address. Set to 4 bytes for an IPv4 address and 16 bytes for an IPv6 address.

AS Number (4 octets): The AS number originating the BGP Remote-Next-Hop attribute and is either a 2-byte AS or 4-Byte AS number

Tunnel Parameter (variable): comprised of multiple sub-TLVs. Each sub-TLV consists of three fields: a 1-octet type, 1-octet length, and zero or more octets of value.

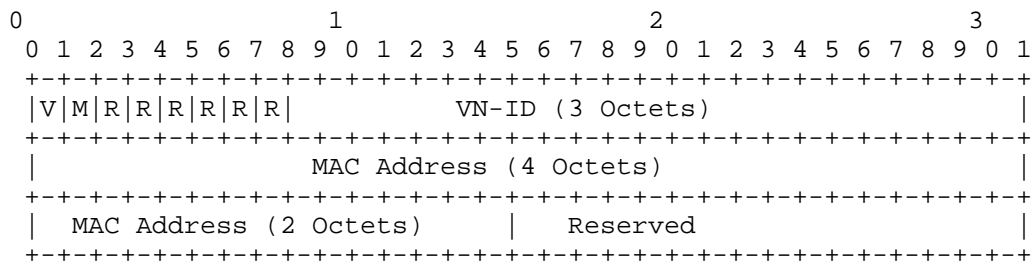
5. Encapsulation sub-TLVs for virtual network overlays

A VN-ID may need to be signaled along with the encapsulation types for DC overlay encapsulations such as [VXLAN] and [NVGRE]. The VN-ID when present in the encapsulation sub-TLV for an overlay encapsulation, MUST be processed by a receiving device if it is capable of understanding it. The details regarding how such a

signaled VN-ID is processed and used is defined in specifications such as [IPVPN-overlay] and [EVPN-overlay].

5.1. Encapsulation sub-TLV for VXLAN

This document defines a new encapsulation sub-TLV format, defined in [RFC5512], for VXLAN tunnels. When the tunnel type is VXLAN, the following is the structure of the value field in the encapsulation sub-TLV:



V: When set to 1, it indicates that a valid VN-ID is present in the encapsulation sub-TLV.

M: When set to 1, it indicates that a valid MAC Address is present in the encapsulation sub-TLV.

R: The remaining bits in the 8-bit flags field are reserved for further use. They MUST be set to 0 on transmit and MUST be ignored on receipt.

VN-ID: Contains a 3 octets VN-ID value, if the 'V' flag bit is set. If the 'V' flag is not set, it SHOULD be set to zero and MUST be ignored on receipt.

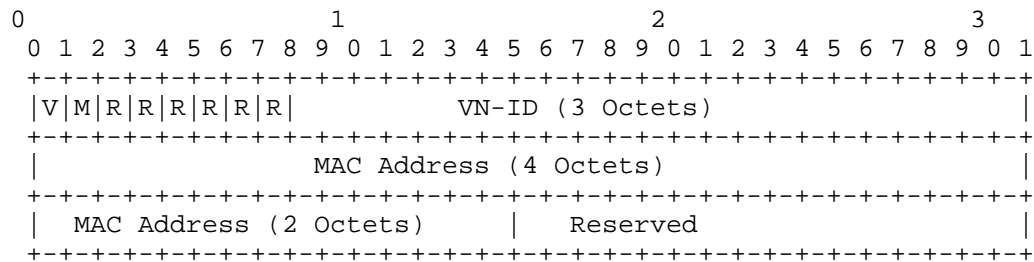
The VN-ID value is filled in the VNI field in the VXLAN packet header as defined in [VXLAN].

MAC Address: Contains a 6 octets of an Ethernet MAC address if the 'M' flag bit is set. If the 'M' flag is not set, it SHOULD set to all zeroes and MUST be ignored on receipt.

The MAC address is local to the device advertising the route, and should be included as the destination MAC address in the inner Ethernet header immediately following the outer VXLAN header, in the packets destined to the advertiser.

5.2. Encapsulation sub-TLV for NVGRE

This document defines a new encapsulation sub-TLV format, defined in [RFC5512], for NVGRE tunnels. When the tunnel type is NVGRE, the following is the structure of the value field in the encapsulation sub-TLV:



V: When set to 1, it indicates that a valid VN-ID is present in the encapsulation sub-TLV.

M: When set to 1, it indicates that a valid MAC Address is present in the encapsulation sub-TLV.

R: The remaining bits in the 8-bit flags field are reserved for further use. They MUST be set to 0 on transmit and MUST be ignored on receipt.

VN-ID: Contains a 3 octets VN-ID value, if the 'V' flag bit is set. If the 'V' flag is not set, it SHOULD be set to zero and MUST be ignored on receipt.

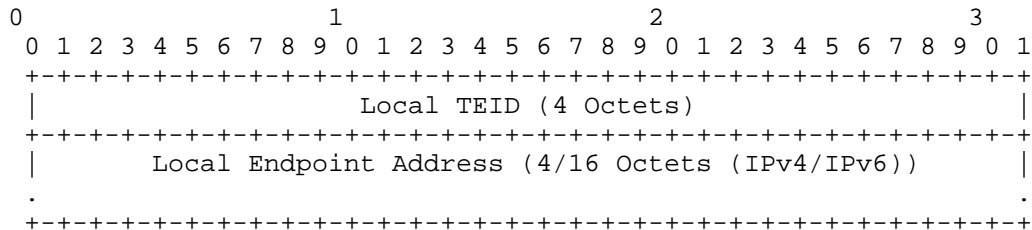
The VN-ID value is filled in the VSID field in the NVGRE packet header as defined in [NVGRE].

MAC Address: Contains a 6 octets of an Ethernet MAC address if the 'M' flag bit is set. If the 'M' flag is not set, it SHOULD set to all zeroes and MUST be ignored on receipt.

The MAC address is local to the device advertising the route, and should be included as the destination MAC address in the inner Ethernet header immediately following the outer NVGRE header, in the packets destined to the advertiser.

5.3. Encapsulation sub-TLV for GTP

This document defines a new encapsulation sub-TLV format, defined in [RFC5512], for GTP tunnels. When the tunnel type is GTP, the following is the structure of the value field in the encapsulation sub-TLV:



Local TEID: Contains a 32-bit Tunnel Endpoint Identifier of a GTP tunnel assigned by EPC that is used to distinguish different connections in received packets within the tunnel.

Local Endpoint Address: Indicates a 4-octets IPv4 address or 16-octets IPv6 address as a local endpoint address of GTP tunnel.

Local Endpoint Address element makes a tunnel endpoint router allow to have multiple Local TEID spaces. Received GTP packets are identified which tunnel connection by combination of Local Endpoint Address and Local TEID.

5.4. Encapsulation for MPLS-in-GRE

This document defines a new encapsulation sub-TLV format, defined in [RFC5512], for MPLS-in-GRE tunnels. When the tunnel type is MPLS-in-GRE, the following is the structure of the value field in an optional encapsulation sub-TLV:


```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               GRE-Key (4 Octets)                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

GRE-Key: 4-octet field [RFC2890] that is generated by the advertising router. The actual method by which the key is obtained is beyond the scope of this document. The key is inserted into the GRE encapsulation header of the payload packets sent by ingress routers to the advertising router. It is intended to be used for identifying extra context information about the received payload. Note that the key is optional. Unless a key value is being advertised, the MPLS-in-GRE encapsulation sub-TLV MUST NOT be present.

Note that signaling a GRE tunnel-type with routes in a labeled SAFI may be sufficient to indicate to the receiver that it needs to send MPLS packets with that GRE encapsulation. However, a specific tunnel-type for MPLS-in-GRE is being defined in order to make this indication explicit to a receiver.

6. Remote-Next-Hop Bestpath Considerations

A BGP speaker SHOULD support a policy to enable the support for using BGP Remote Nexthop attribute. An implementation that supports the BGP Remote-Next-Hop MUST use BGP Nexthop attribute information whenever BGP Remote-Next-Hop is not enabled.

Traditionally a BGP speaker uses the IGP cost towards the BGP Next-Hop as a BGP path selection criteria. However, when a BGP speaker is configured to use the BGP Remote-Next-Hop value, then it SHOULD use the IGP cost towards the IP address selected from the Remote-Next-Hop attribute. When there are multiple such IP addresses that may be installed, it SHOULD use the worst IGP cost among them.

Similarly, the speaker SHOULD also check that the IP address is reachable before considering that path eligible for bestpath.

7. Securing Remote-Next-Hop

The Remote-Next-Hop attribute provides a set of tunnel parameters. While the Remote-Next-Hop attribute has as goal to inform an intended recipient with these tunnel parameters, it is important to make sure that the attributes have not been tampered with and that they are restricted to the intended scope of distribution for secure operation.

7.1. Restrictions on Announcing of Remote-Next-Hop Attribute

The Remote-Next-Hop attribute is used to carry an additional information (tunnel end-point, encapsulation type, etc). It has a security value to contain the distribution of the Remote-Next-Hop attribute within its planned scope of distribution. This scope could be, but is not limited to, a particular department, site, organization, across ASes within a same administration control or a global scope.

To contain distribution of the Remote-Next-Hop attribute beyond its intended scope of applicability, attribute filtering MAY be deployed. The BGP speaker communicating to a speaker beyond the intended scope of the Remote-Next-Hop attribute SHOULD filter the attribute during the route announcements.

To facilitate attribute filtering, an implementation that supports the BGP Remote-Next-Hop attribute MUST support a policy to (1) ignore the received attribute and (2) filter the attribute.

7.2. Restrictions on Originating of Remote-Next-Hop Attribute

A BGP Remote-Next-Hop attribute may be added to routes that belong to same Autonomous system as the tunnel endpoint address. Implementations SHOULD validate the following to ensure the validity of Remote-Next-Hop Attribute:

(1) BGP Remote-Next-Hops Tunnel Endpoint and AS number association SHOULD be validated using BGP Origin Validation.

(2) BGP Remote-Next-Hop Tunnel Endpoints underlay routes origin AS SHOULD be validated using BGP Origin Validation. This AS number MUST be the same as the AS number carried within BGP Remote-Next-Hop attribute.

(3) The origin AS of BGP Routes that carry BGP Remote-Next-Hop attribute SHOULD be validated using BGP Origin Validation. This AS number MUST be same as the AS number carried within BGP Remote-Next-Hop attribute.

If the above validation fails, the tunnel type SHOULD be considered as invalid. This does not affect the validity of the others tunnels types carried within the Remote-Next-Hop Attribute.

8. Multiple tunnel endpoint addresses

In some cases, a device may need to accept incoming traffic for a prefix via multiple different encapsulations, to support interactions with remote devices with disjoint capabilities. Certain device implementations cannot support the use of the same IP address as local tunnel endpoint for multiple encapsulations.

In certain cases, a device may need to signal an additional, alternate tunnel endpoint address, to be used by other devices only as a backup in certain failure conditions.

9. Attribute error handling

When receiving a BGP Update message containing a malformed Remote-Next-Hop attribute, the attribute MUST be quietly ignored and not passed along to other BGP peers. (see [draft-ietf-idr-error-handling], Section 7). This is equivalent to the -attribute discard-action specified in [draft-ietf-idr-error-handling]. An implementation MAY log an error for further analysis.

Note that a BGP Remote-Next-Hop attribute MUST NOT be considered to be malformed because it contains more than one TLV of a given type or because it contains TLVs of unknown types.

If a BGP path attribute is received that has the Remote-Next-Hop attribute codepoint but does not have the transitive bit set, the attribute MUST be considered to be a malformed Remote-Next-Hop attribute and MUST be discarded as specified in this section.

10. BGP speakers that do not support BGP Remote-Next-Hop attribute

If a BGP Speaker does not support this attribute, and receives this attribute, then it follows the normal NLRI processing and BGP best path selection, and the resulting forwarding decision is used, as the attribute is optional.

11. Use Case scenarios

This section provides a brief overview of some use-cases for the BGP Remote-Next-Hop attribute. Use of the BGP Remote-Next-Hop is not limited to the examples in this section. Details regarding how the attribute is used are described in the respective solution drafts that are referenced where necessary.

11.1. Stateless user-plane architecture for virtualized EPC (vEPC)

The full usage case of BGP Remote-Next-Hop regarding vEPC can be found in [vEPC], while [RFC6459] documents IPv6 in 3GPP EPS.

3GPP introduces Evolved Packet Core (EPC) that is fully IP based mobile system for LTE and -advanced in their Release-8 specification and beyond. Operators are now deploying EPC for LTE services and encounter rapid LTE traffic growth. There are various activities to offload mobile traffic in 3GPP and IETF such as LIPA, SIPTO and DMM. The concept is similar that traffic of OTT (Over The Top) application is offloaded at entity that is closer to the mobile node (ex. eNodeB or closer anchor).

11.2. Stateless User-plane Architecture for virtual Packet Edge

With the emergence of the NfV technologies, different architectures are proposed for virtualized services. These functions will normally run in the datacenter. BGP remote-next-hop can be used to inject traffic into the virtualized services running in the datacenter using tunnels. These tunnels can be signalled using BGP remote-next-hop. This facilitates a dynamic, simple and clean routing architecture. BGP Remote Next Hop can simplify the orchestration or provisioning layer by signalling the tunnel endpoint (virtual provider edge router) in combination with the encapsulation protocol.

If this is used together with orchestrated traffic steering mechanisms (i.e. BGP Flowspec) , it is possible to differentiate at application level, and forward each different traffic types towards the desired destination.

11.3. Dynamic Network Overlay Infrastructure

The BGP Remote-Next-Hop extension allows consistent signalling of tunnel encapsulations as needed by virtual network overlay solutions such as [I-D.drao-bgp-l3vpn-virtual-network-overlays] and [I-D.sd-l2vpn-evpn-overlay]

11.4. Simple VPN solution using Multi-point Security Association

[draft-yamaya-ipsecme-mps] describes the overlay network solution by utilizing dynamically established IPsec multi-point Security Association (SA) without individual connection.

Multi-point SA technology provides the simplified mechanism of the Auto Discovery and Configuration function. This is applicable for any IPsec tunnels such as IPv4 over IPv4, IPv4 over IPv6, IPv6 over IPv4 and IPv6 over IPv6.

MPSA does not provide peer discovery function by itself. However, other mechanism, such as BGP, can be employed with MPSA for automatic peer discovery. BGP Remote-Next-Hop can be used to learn peer information as next-hops.

12. IANA Considerations

This document defines a new BGP attribute known as a BGP Remote-Next-Hop attribute. We request IANA to allocate a new attribute code from the -BGP Path Attributes- registry with a symbolic name -Remote-Next-Hop- attribute.

We also request IANA to allocate four new BGP Tunnel Types from the -BGP Tunnel Encapsulation Attribute Tunnel Types- registry with the following symbolic names: -VXLAN- with Tunnel type 8, -NVGRE- with Tunnel type 9, -GTP- with Tunnel type 10, -MPLS-in-GRE with Tunnel type 11, -MPLS-in-UDP- with Tunnel type 12 and -MPLS-in-UDP-with-DTLS with Tunnel type 13.

13. Security Considerations

This technology could be used as technology as man in the middle attack, however with existing RPKI validation for BGP that risk is reduced.

The distribution of Tunnel end-point address information can result in potential DoS attacks. Therefore is it strongly recommended to install traffic filters, IDSs and IPSs at the perimeter of the tunneled network infrastructure.

measures SHOULD be taken to protect the validity of the BGP Remote-Next-Hop attribute. It is possible to inject a rogue BGP Remote-Next-Hop attribute to an NLRI resulting in Monkey-In-The-Middle attack (MITM). To avoid this type of MITM attack, it is strongly recommended to use a technology mechanism to verify that for NLRI it is the expected BGP Remote-Next-Hop. We anticipate that this can be done with an expansion of RPKI-Based origin validation, see [I-D.ietf-sidr-pfx-validate].

This does not avoid the fact that rogue AS numbers may be inserted or injected into the AS-Path. To achieve protection against that threat BGP Path Validation should be used, see [I-D.ietf-sidr-bgpsec-overview].

14. Privacy Considerations

This proposal may introduce privacy issues, however with BGP security mechanisms in place they should be prevented.

15. Acknowledgements

The authors would like to thanks Satoru Matsushima, Bruno Decraene, Ryuji Wakikawa and Miya Kohno for their usefull vEPC discussions. Istvan Kakonyi provided insight in the vPE use case scenario.

Satoshi Usui provided datapoints around Simple VPN solution using Multi-point Security Association.

16. Change Log

Initial Version: 16 May 2012

Hacked for -01: 17 July 2012

Hacked for -05: 07 January 2014

Hacked for -07: 15 September 2014

17. References

17.1. Normative References

- [I-D.ietf-mpls-in-udp]
Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black,
"Encapsulating MPLS in UDP", draft-ietf-mpls-in-udp-11
(work in progress), January 2015.
- [RFC1771] Rekhter, Y. and T. Li, "A Border Gateway Protocol 4 (BGP-4)", RFC 1771, March 1995.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.
- [RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.
- [RFC3931] Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.

- [RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC5512] Mohapatra, P. and E. Rosen, "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009.
- [RFC5566] Berger, L., White, R., and E. Rosen, "BGP IPsec Tunnel Encapsulation Attribute", RFC 5566, June 2009.
- [RFC6459] Korhonen, J., Soeninen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, January 2012.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014.

17.2. Informative References

- [I-D.drao-bgp-l3vpn-virtual-network-overlays]
Rao, D., Mullooly, J., and R. Fernando, "Layer-3 virtual network overlays based on BGP Layer-3 VPNs", draft-drao-bgp-l3vpn-virtual-network-overlays-03 (work in progress), July 2014.
- [I-D.ietf-idr-error-handling]
Chen, E., Scudder, J., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", draft-ietf-idr-error-handling-13 (work in progress), June 2014.
- [I-D.ietf-sidr-bgpsec-overview]
Lepinski, M., "An Overview of BGPsec", draft-ietf-sidr-bgpsec-overview-06 (work in progress), January 2015.
- [I-D.ietf-sidr-pfx-validate]
Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", draft-ietf-sidr-pfx-validate-10 (work in progress), October 2012.

- [I-D.matsushima-stateless-uplane-vepc]
Matsushima, S. and R. Wakikawa, "Stateless user-plane architecture for virtualized EPC (vEPC)", draft-matsushima-stateless-uplane-vepc-01 (work in progress), July 2013.
- [I-D.sd-l2vpn-evpn-overlay]
Sajassi, A., Drake, J., Bitar, N., Isaac, A., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution using EVPN", draft-sd-l2vpn-evpn-overlay-03 (work in progress), June 2014.
- [I-D.sridharan-virtualization-nvgre]
Sridharan, M., Greenberg, A., Wang, Y., Garg, P., Venkataramiah, N., Duda, K., Ganga, I., Lin, G., Pearson, M., Thaler, P., and C. Tumuluri, "NVGRE: Network Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre-05 (work in progress), July 2014.
- [I-D.yamaya-ipsecme-mps-a]
Yamaya, A., Ohya, T., Yamagata, T., and S. Matsushima, "Simple VPN solution using Multi-point Security Association", draft-yamaya-ipsecme-mps-a-04 (work in progress), July 2014.

Authors' Addresses

Gunter Van de Velde

Email: gunter@vandevelde.cc

Keyur Patel
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95124 95134
USA

Email: keyupate@cisco.com

Dhananjaya Rao
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95124 95134
USA

Email: dhrao@cisco.com

Robert Raszuk
NTT MCL Inc.
101 S Ellsworth Avenue Suite 350
San Mateo, CA 94401
US

Email: robert@raszuk.net

Randy Bush
Internet Initiative Japan
5147 Crystal Springs
Bainbridge Island, Washington 98110
US

Email: randy@psg.com

Network Working Group
Internet-Draft
Updates: RFC 4724 (if approved)
Intended status: Standards Track
Expires: April 26, 2013

H. Zhang
HangZhou H3C Co. Limited
A. Retana
Cisco Systems, Inc.
October 22, 2012

Transitive BGP Graceful Restart
draft-zhang-idr-transitive-gr-01

Abstract

This document defines an extension to BGP Graceful Restart that reduces the negative impact of multiple inter-connected routers restarting. The proposed mechanism does not require any changes to the BGP protocol.

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 26, 2013.

Copyright Notice

Copyright (c) 2012 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. Requirements Language | 3 |
| 3. Proposed Solution | 4 |
| 4. Security Considerations | 4 |
| 5. IANA Considerations | 5 |
| 6. Acknowledgements | 5 |
| 7. References | 5 |
| 7.1. Normative References | 5 |
| 7.2. Informative References | 5 |
| Authors' Addresses | 5 |

1. Introduction

The BGP Graceful Restart [RFC4724] process defines a mechanism that a restarting router can use with its non-restarting peers. The existence of other restarting routers results in the use of the base route exchange mechanism [RFC4271] with them, even if the forwarding state has indeed been preserved for (and by) those peers during the restart. As a result, traffic forwarding between restarting routers is disrupted.

This document defines an extension to BGP Graceful Restart that reduces the negative impact of multiple inter-connected restarting routers. The proposed mechanism does not require any changes to the BGP protocol.

The current process [RFC4724] states that routes from restarting peers are to be removed from the local forwarding state when the non-restarting peers converge (the End-of-RIB marker is received from all of them). Assuming a simple topology:

NR1 - R2 - R3 - NR4

where NRx are non-restarting routers, Rx are restarting routers and the lines between them represent BGP sessions.

There are two types of routes affected (from R2's point of view) by the current process:

1. Routes that are only reachable through R3. These routes will be removed from the forwarding table when the non-restarting routers converge, and installed back in when the convergence with R3 is done.
2. Routes that are reachable through both R3 and NR1. These routes will first change to NR1 when the non-restarting routers converge, and later back to R3 (assuming that is in fact still the preferred path).

Both types can clearly cause disruption in traffic forwarding, micro-loops, traffic loss, etc.

2. Requirements Language

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

3. Proposed Solution

The extension proposed to BGP Graceful Restart to accommodate for multiple restarting routers, when the forwarding state has been preserved between them, is simply to delay sending the End-of-RIB marker to non-restarting routers.

Specifically, to allow a restarting router the ability to reduce the impact due to other restarting routers, the following paragraph is added as the fifth one in section 4.1 (Procedures for the Restarting Speaker) [RFC4724]:

Before updating the corresponding forwarding states, the BGP speaker MAY advertise the Adj-RIB-Out to the remaining peers (ones with the "Restart State" bit set in the received capability and ones that do not advertise the graceful restart capability), including the End-of-RIB marker, and MAY wait for the corresponding End-of-RIB marker from the restarting ones.

During the recovery period of multiple restarting routers, a BGP speaker may advertise routing information that is not being used at the time. Because the forwarding state of the speakers remains unchanged (from that at the restart), it is clear that this transitive property of sharing routing information between restarting routers doesn't cause any issues in the actual forwarding of traffic. Furthermore, it has the advantage of avoiding further disruptions in the forwarding of traffic through the restarting routers.

In order to maintain the transitive property when more than two BGP speakers peering with each other restart, the following paragraph is added as the sixth one in section 4.1 (Procedures for the Restarting Speaker) [RFC4724]:

If a restarting BGP speaker has multiple restarting peers, sending the End-of-RIB marker SHOULD be delayed until all the markers from restarting peers with a lower BGP Identifier have been received. The BGP speaker with the lowest BGP Identifier on a given connection SHOULD send its End-of-RIB marker if the pair hasn't sent or received UPDATES for a locally configured time period (which should be significantly less than the Selection_Deferral_Timer).

4. Security Considerations

This document proposes an extension to an existing mechanism. The same security considerations explained there apply to this extension.

The propagation of routing information that is not in use may cause forwarding loops and an inconsistent state in a network. However, the risk in this document is mitigated by the fact that the information is validated by all peers once the convergence process completes.

5. IANA Considerations

This document has no IANA actions.

6. Acknowledgements

The authors would like to thank Enke Chen, John Scudder, Robert Raszuk and Abhay Roy for their feedback.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4724] Sangli, S., Chen, E., Fernando, R., Scudder, J., and Y. Rekhter, "Graceful Restart Mechanism for BGP", RFC 4724, January 2007.

7.2. Informative References

- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

Authors' Addresses

Haifeng Zhang
Hangzhou H3C Co. Limited
310 Liuhe Road, Zhijiang Science Park
Hangzhou
P.R. China

Email: zhanghf@h3c.com

Alvaro Retana
Cisco Systems, Inc.
7025 Kit Creek Rd.
Research Triangle Park, NC 27709
USA

Email: aretana@cisco.com

