BIER
Internet-Draft
Intended status: Standards Track
Expires: January 7, 2020

Z. Zhang
Juniper Networks
N. Warnke
Deutsche Telekom
I. Wijnands
Cisco Systems
D. Awduche
Verizon

July 6, 2019

Tethering A BIER Router To A BIER-incapable Router
draft-zzhang-bier-tether-02

Abstract

This document specifies optional procedures to optimize the handling of Bit Index Explicit Replication (BIER) incapable routers, by tethering a BIER router to a BIER incapable router.

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.

Table of Contents

1. Terminologies

Familiarity with BIER architecture, protocols and procedures is assumed. Some terminologies are listed below for convenience.

[To be added].

2. Introduction

Consider the following scenario where router X does not support BIER.

```
                           ------ BFR2 ------- BFER2
                          /
   BFER1 ---  BFR1 ---- X ------- BFR3 ------- BFER3
                          .........
                          \
                           ------ BFRn ------- BFERn
```

For BFR1 to forward BIER traffic towards BFR2...BFRn, it needs to tunnel individual copies through X. This degrades to "ingress" replication to those BFRs. If X's connections to BFRs are long

distance or bandwidth limited, and n is large, it becomes very inefficient.

A solution to the inefficient tunneling from BFRs is to tether a BFRx to X:

```
                          ------ BFR2 ------- BFER2
                         /
BFER1 ---  BFR1 ---- X ------- BFR3 ------- BFER3
                    / \  .........
                   /   \
                BFRx    ------ BFRn ------- BFERn
```

Instead of BFR1 tunneling to BFR2, ..., BFRn directly, BFR1 will get BIER packets to BFRx, who will then tunnel to BFR2, ..., BFRn. There could be fat and local pipes between the tethered BFRx and X, so ingress replication from BFRx is acceptable.

For BFR1 to tunnel BIER packets to BFRx, the BFR1-BFRx tunnel need to be announced in IGP as a forwarding adjacency so that BFRx will appear on the SPF tree. This need to happen in a BIER specific topology so that unicast traffic would not be tunneled to BFRx. Obviously this is operationally cumbersome.

Section 6.9 of BIER architecture specification [RFC8279] describes a method that tunnels BIER packets through incapable routers without the need to announce tunnels. However that does not work here, because BFRx will not appear on the SPF tree of BFR1.

There is a simple solution to the problem though. Even though X does not support BIER forwarding, it could advertises BIER information as if it supported BIER so BFRs will send BIER packets to it. The BIER packets have a BIER label in front of the BIER header and X will use the BIER label to label switch to BFRx, who will in turn do BIER forwarding to other BFRs but via tunneling as described in section 6.9 of BIER architecture spec.

Even though X advertises as if it supported BIER, BFRx needs to know that X does not really support BIER so it will tunnel to other BFRs through X. The knowledge is through static provisioning or through additional signaling. In the latter case, X could advertise that BFRx is its helper node, so that other BFRs could optionally use the Section 6.9 method to tunnel to BFRx, instead of sending native BIER packets to X and rely on X label switching to BFRx. This also allows it to work in the non-MPLS case.

Alternatively, instead of for X to advertise that it supports BIER but relies on helper BFRx, BFRx could advertise that it is X's helper and other BFRs will use BFRx (instead of X's children on the SPF tree) to replace X during its post-SPF processing as described in section 6.9 of BIER architecture spec. That way, X does not need any special knowledge, provisioning or procedure.

The two options both have pros and cons - the first option only needs X and BFRx to support the new procedure while the second option does not require anything to be done to the BIER incapable X.

BFRx could also be connected to other routers in the network so that it could send BIER packets through other routers as well, not necessarily tunneling through X. To prevent routing loops, smallest metric, which is 1, must be announced for links between X and BFRx in both directions.

3. Additional Considerations

While the example shows a local connection between BFRx and X, it does not have to be like that. As long as packets can arrive at BFRx without requiring X to do BIER forwarding, it should work. For example, X could label switch incoming BIER packets through a multi-hop tunnel to BFRx, or other BFRs could tunnel BIER packets to BFRx based on X's advertisement that BFRx is its helper. However, BFRx must make sure that if X appears in its SPF paths to some BFERs, then it must tunnel BIER packets for those BFERs directly to X's BFR children on BFRx's SPF tree.

Additionally, the helper BRFx can be a transit helper, i.e., it has other connections (instead of being a stub helper that is only connected to X), as long as BFRx won't send BIER packets tunneled to it back towards the tunnel ingress:

```
                              ------ BFR2 ------- BFER2
                             /
      BFER1 ---  BFR1 ---- X ------- BFR3 ------- BFER3
                           |
                           |
                         BFRx ------ BFR4 ------- BFER4
                             \
                              ------ BFR5 ------- BFER5
```

In the following example, there is a connection between BFR1 and BFRx. If the link metrics are all 1 on the three sides of BFR1-X-BFRx triangle, loop won't happen but if the BFRx-X metric is 3 while other two sides of the triangle has metric 1 then BFRx will

send BIER packets tunneled to it from BFR1 back to BFR1, causing a loop.

```
                            ------ BFR2 ------- BFER2
                           /
   BFER1 ---  BFR1 ---- X ------- BFR3 ------- BFER3
                \      / \  .........
                 \    /   \
                   BFRx    ------ BFRn ------- BFERn
```

This can easily be prevented if BFR1 does an SPF calculation with the helper BFRx as the root. For any BFERn reached via X from BFR1, if BFRx's SPF path to BFERn includes BFR1 then BFR1 must not use the helper. Instead, BFR1 must directly tunnel packets for BFERn to X's BFR (grand-)child on BFR1's SPF path to BFERn, per section 6.9 of [RFC8279].

Notice that this SPF calculation on BFR1 with BFRx as the root is no different from the SPF done for a neighbor as part of LFA calculation. In fact, BFR1 tunneling packets to X's helper is no different from sending packets to a LFA backup.

Also notice that, instead of a dedicated helper BFRx, any one or multiple ones of BFR2..N can also be the helper (as long as the connection between that BFR and X has enough bandwidth for replication to multiple helpers through X). To allow multiple helpers to help the same non-BFR, the "I am X's helper" advertisement carries a priority. BFR1 will choose the helper advertising the highest priority among those satisfying the loop-free condition described above. When there are multiple helpers advertising the same priority and satisfying the loop-free condition, any one or multiple ones could be used solely at the discretion of BFR1. However, if multiple ones are used, it means that multiple copies may be tunneled through X.

The following situation where a helper BFRxy helps two different non-BFRs X and Y also works. It's just a special situation of a transit helper.

```
                              ----- BFR2 ------- BFER2
                            /
                          X ------- BFR3 ------- BFER3
                        / | \
                      /     \  ----- BFR4 ------- BFER4
                    /         \
          BFIR1 -- BFR1      BFRxy ------------- BFERxy
                    \         /
                      \     /  ----- BFR5 ------- BFER5
                        \ | /
                          Y ------- BFR6 ------- BFER6
                            \
                              ----- BFRn ------- BFERn
```

4. Specification

The procedures in this document apply when a BFRx is tethered to a BIER incapable router X as X's helper for BIER forwarding.

BFRx MUST not send BIER packets natively to X even if X advertises BIER information. BFRx knows that X does not really support BIER either from provisioning or from the BIER Helper Node sub-sub-TLV advertised by X.

Procedures for BGP signaling is described in Section 4.3.

Either of the following two methods may be used for ISIS [RFC8401] and OSPF [RFC8444]. The sub-sub-TLVs for both methods have the same format: the value is BIER prefix of the helper/helped node followed by a one-octet priority field, and one-octet reserved field. The length is 6 for IPv4 and 18 for IPv6 respectively.

4.1. Advertising from Helped Node

For non-MPLS encapsulation, X MUST advertise a BIER Helper Node sub-sub-TLV that specifies the BIER prefix of the helper BFRx. Other BFRs MUST use the Section 6.9 procedure modified as following: X is treated as BIER incapable (because of the BIER Helper Node sub-sub-TLV), and is replaced with the BFRx (instead of X's children on the SPF tree) during the post-SPF processing.

This requires other BFRs to recognize the BIER Helper Node sub-sub-TLV. The same procedure MAY be used For MPLS encapsulation, though with the following alternative for MPLS encapsulation, tethering is transparent to other BFRs (except the helper node BFRx) - they do not need to be aware that X does not support BIER at all.

For MPLS encapsulation, X MAY advertises BIER information as if it supported BIER forwarding, including the MPLS Encapsulation sub-sub-TLV with a label range. X MUST set up its forwarding state such that incoming packets with a BIER label in its advertised label range are label switched to BFRx, either over a direct link or through a tunnel. The incoming label is swapped to a BIER label advertised by BFRx for the <sub-domain, bsl, set> that the incoming label corresponds to.

Notice that both methods can be used for MPLS encapsulation at the same time. In that case another BFR may send BIER packets to X natively, or tunnel to BFRx directly.

4.2. Advertising from Helper Node

With this method, the helper node (BFRx) MUST advertise a BIER Helped Node sub-sub-TLV that specifies the BIER incapable node (X) that this node helps. When other BFRs follow the post-SPF processing procedures as specified in section 6.9 of the BIER architecture spec [RFC8279], they replace the helped node on the SPF tree with the helper node (instead of the children of the helped node).

4.3. Procedures for BGP Signaling

Suppose that the BIER domain uses BGP signaling [I-D.ietf-bier-idr-extensions] instead of IGP. BFR1..N advertises BIER prefixes that are reachable through them, with BIER Path Attributes (BPA) attached. There are three situations regarding X's involvement:

(1) X does not participate in BGP peering at all

(2) X re-advertises the BIER prefixes but does not do next-hop-self

(3) X re-advertises the BIER prefixes and does next-hop-self

With (1) and (2), the BFR1..N will tunnel BIER packets directly to each other. It works but not efficiently as explained earlier. With (3), BIER forwarding will not work, because BFR1..N would try to send BIER packets to X though X does not advertise any BIER information. If Tunnel Encapsulation Attribute (TEA) [I-D.ietf-idr-tunnel-encaps] is used as specified in [I-D.zzhang-bier-multicast-as-a-service] with (3), then it becomes similar to (2) - works but still not efficiently.

To make tethering work well with BGP signaling, the following can be done:

- Configure a BGP session between X and its helper BFRx. X re-advertises BIER prefixes (with BPA) to BFRx without changing the tunnel destination address in the TEA.

- BFRx advertises its own BIER prefix with BPA to X, and sets the tunnel destination address in the TEA to itself. X then re-advertises BFRx's BIER prefix to BFR1..N, without changing the tunnel destination address in the TEA.

- For BIER prefixes (with BIER Path Attribute) that X re-advertises to other BFRs, the tunnel destination in the TEA is changed to the helper BFRx.

With the above, BFR1..N will tunnel BIER packets to BFRx (following the tunnel destination address in the TEA), who will then tunnel packets to other BFRs (again following the tunnel destination address in the TEA). Notice that what X does is not specific to BIER at all.

5. Security Considerations

This specification does not introduce additional security concerns beyond those already discussed in BIER architecture and OSPF/ISIS/BGP extensions for BIER signaling.

6. IANA Considerations

This document requests two new sub-sub-TLV type values from the "Sub-sub-TLVs for BIER Info Sub-TLV" registry in the "IS-IS TLV Codepoints" registry:

Type	Name
TBD1	BIER Helper Node
TBD2	BIER Helped Node

This document also requests two new sub-TLV type values from the OSPFv2 Extended Prefix TLV Sub-TLV registry:

Type	Name
TBD3	BIER Helper Node
TBD4	BIER Helped Node

7. Acknowledgements

The author wants to thank Eric Rosen and Antonie Przygienda for their review, comments and suggestions.

8. Normative References

[I-D.ietf-bier-idr-extensions]
Xu, X., Chen, M., Patel, K., Wijnands, I., and T. Przygienda, "BGP Extensions for BIER", draft-ietf-bier-idr-extensions-06 (work in progress), January 2019.

[I-D.ietf-idr-tunnel-encaps]
Patel, K., Velde, G., Ramachandra, S., and E. Rosen, "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-12 (work in progress), May 2019.

[I-D.zzhang-bier-multicast-as-a-service]
Zhang, Z., Rosen, E., and L. Geng, "Multicast/BIER As A Service", draft-zzhang-bier-multicast-as-a-service-00 (work in progress), October 2018.

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

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8401] Ginsberg, L., Ed., Przygienda, T., Aldrin, S., and Z. Zhang, "Bit Index Explicit Replication (BIER) Support via IS-IS", RFC 8401, DOI 10.17487/RFC8401, June 2018, <https://www.rfc-editor.org/info/rfc8401>.

[RFC8444] Psenak, P., Ed., Kumar, N., Wijnands, IJ., Dolganow, A., Przygienda, T., Zhang, J., and S. Aldrin, "OSPFv2 Extensions for Bit Index Explicit Replication (BIER)", RFC 8444, DOI 10.17487/RFC8444, November 2018, <https://www.rfc-editor.org/info/rfc8444>.

Authors' Addresses

Zhaohui Zhang
Juniper Networks

EMail: zzhang@juniper.net

Nils Warnke
Deutsche Telekom

EMail: Nils.Warnke@telekom.de

IJsbrand Wijnands
Cisco Systems

EMail: ice@cisco.com

Daniel Awduche
Verizon

EMail: daniel.awduche@verizon.com