INTERNET-DRAFT
Intended Status: Standard Track

Sami Boutros
VMware
Ali Sajassi
Cisco Systems
John Drake
Juniper Networks
Jorge Rabadan
Nokia
Sam Aldrin
Google

Expires: September 7, 2019 March 6, 2019

EVPN control plane for Geneve
draft-boutros-bess-evpn-geneve-04.txt

Abstract

This document describes how Ethernet VPN (EVPN) control plane can be used with Network Virtualization Overlay over Layer 3 (NVO3) Generic Network Virtualization Encapsulation (Geneve) encapsulation for NVO3 solutions. EVPN control plane can also be used by a Network Virtualization Endpoints (NVEs) to express Geneve tunnel option TLV(s)supported in transmission and/or reception of Geneve encapsulated data packets.

Status of this Memo

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

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

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

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

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

Table of Contents

1 Introduction

The Network Virtualization over Layer 3 (NVO3) solutions for network virtualization in data center (DC) environment are based on an IP-based underlay. An NVO3 solution provides layer 2 and/or layer 3 overlay services for virtual networks enabling multi-tenancy and workload mobility. The NVO3 working group have been working on different dataplane encapsulations. The Generic Network Virtualization Encapsulation [GENEVE] have been recently recommended to be the proposed standard for network virtualization overlay encapsulation.

This document describes how the EVPN control plane can signal Geneve encapsulation type in the BGP Tunnel Encapsulation Extended Community defined in [TUNNEL-ENCAP]. In addition, this document defines how to communicate the Geneve tunnel option types in a new BGP Tunnel Encapsulation Attribute sub-TLV. The Geneve tunnel options are encapsulated as TLVs after the Geneve base header in the Geneve packet as described in [GENEVE].

[DT-ENCAP] recommends that a control plane determines how Network Virtualization Edge devices (NVEs) use the GENEVE option TLVs when sending/receiving packets. In particular, the control plane negotiates the subset of option TLVs supported, their order and the total number of option TLVs allowed in the packets. This negotiation capability allows, for example, interoperability with hardware-based NVEs that can process fewer options than software-based NVEs.

This EVPN control plane extension will allow a Network Virtualization Edge (NVE) to express what Geneve option TLV types it is capable to receive or to send over the Geneve tunnel to its peers.

In the datapath, a transmitting NVE MUST NOT encapsulate a packet destined to another NVE with any option TLV(s) the receiving NVE is not capable of processing.

1.1 Terminology

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

Most of the terminology used in this documents comes from [RFC7432] and [NVO3-FRWK].

NVO3: Network Virtualization Overlay over Layer 3

GENEVE: Generic Network Virtualization Encapsulation.

NVE: Network Virtualization Edge.

VNI: Virtual Network Identifier.

MAC: Media Access Control.

OAM: Operations, Administration and Maintenance.

PE: Provide Edge Node.

CE: Customer Edge device e.g., host or router or switch.

EVPN: Ethernet VPN.

EVI: An EVPN instance spanning the Provider Edge (PE) devices participating in that EVPN.

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on a PE.

2. GENEVE extensions

This document adds some extensions to the [GENEVE] encapsulation that are relevant to the operation of EVPN.

2.1 Ethernet option TLV

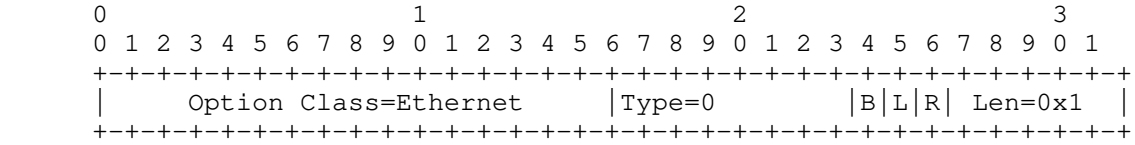
[EVPN-OVERLAY] describes when an ingress NVE uses ingress replication to flood unknown unicast traffic to the egress NVEs, the ingress NVE needs to indicate to the egress NVE that the Encapsulated packet is a BUM traffic type. This is required to avoid transient packet duplication in all-active multi-homing scenarios. For GENVE encapsulation we need a bit to for this purpose.

[RFC8317] uses MPLS label for leaf indication of BUM traffic originated from a leaf AC in an ingress NVE so that the egress NVEs can filter BUM traffic toward their leaf ACs. For GENVE encapsulation we need a bit for this purpose.

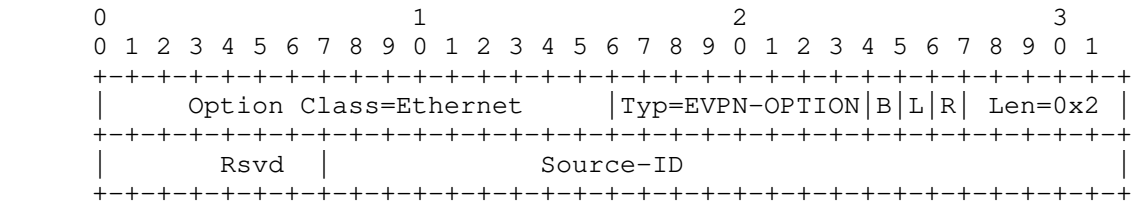
Although the default mechanism for split-horizon filtering of BUM traffic on an Ethernet segment for IP-based ecnapsulations such as VxLAN, GPE, NVGRE, and GENVE, is local-bias as defined in section 8.3.1 of [EVPN-OVERLAY], there can be an incentive to leverage the same split-horizon filtering mechanism of [RFC7432] that uses a 20-bit MPLS label so that a) the a single filtering mechanism is used for all encapsulation types and b) the same PE can participate in a mix of MPLS and IP encapsulations. For this purpose a 20-bit label

field MAY be defined for GENVE encapsulation. The support for this label is optional.

If an NVE wants to use local-bias procedure, then it sends the new option TLV without ESI-label (e.g., length=4):



If an NVE wants to use ESI-label, then it sends the new option TLV with ESI-label (e.g., length=8)



Where:

- Option Class is set to Ethernet (new Option Class requested to IANA)
- Type is set to EVPN-OPTION (new type requested to IANA) and C bit must be set.
- B bit is set to 1 for BUM traffic.
- L bit is set to 1 for Leaf-Indication.
- Source-ID is a 24-bit value that encodes the ESI-label value signaled on the EVPN Autodiscovery per-ES routes, as described in [RFC7432] for multi-homing and [RFC8317] for leaf-to-leaf BUM filtering. The ESI-label value is encoded in the high-order 20 bits of the Source-ESI field.

The egress NVEs that make use of ESIs in the data path (because they have a local multi-homed ES or support [RFC8317]) SHOULD advertise their Ethernet A-D per-ES routes along with the Geneve tunnel sub-TLV and in addition to the ESI-label Extended Community. The ingress NVE can then use the Ethernet option-TLV when sending GENEVE packets based on the [RFC7432] and [RFC8317] procedures. The egress NVE will use the Source-ID field in the received packets to make filtering decisions.

Note that [EVPN-OVERLAY] modifies the [RFC7432] split-horizon procedures for NVO3 tunnels using the "local-bias" procedure. "Local-

bias" relies on tunnel IP source address checks (instead of ESI-labels) to determine whether a packet can be forwarded to a local ES.

While "local-bias" MUST be supported along with GENEVE encapsulation, the use of the Ethernet option-TLV is RECOMMENDED to follow the same procedures used by EVPN MPLS.

An ingress NVE using ingress replication to flood BUM traffic MUST send B=1 in all the GENEVE packets that encapsulate BUM frames. An egress NVE SHOULD determine whether a received packet encapsulates a BUM frame based on the B bit. The use of the B bit is only relevant to GENEVE packets with Protocol Type 0x6558 (Bridged Ethernet).

3. BGP Extensions

As per [EVPN-OVERLAY] the BGP Encapsulation extended community defined in [TUNNEL-ENCAP] is included with all EVPN routes advertised by an egress NVE.

This document specifies a new BGP Tunnel Encapsulation Type for Geneve and a new Geneve tunnel option types sub-TLV as described below.

3.1 Geneve Tunnel Option Types sub-TLV

The Geneve tunnel option types is a new BGP Tunnel Encapsulation Attribute Sub-TLV.

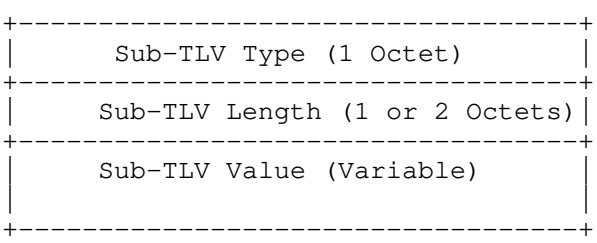
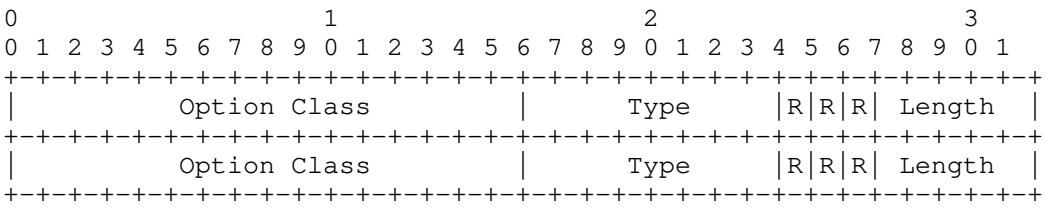


Figure 1: Geneve tunnel option types sub-TLV

The Sub-TLV Type field contains a value in the range from 192-252. To be allocated by IANA.

Sub-TLV value MUST match exactly the first 4-octets of the option TLV format. For instance, if we need to signal support for two option TLVs:



Where, an NVE receiving the above sub-TLV, will send GENEVE packets to the originator NVE with with only the option TLVs the receiver NVE is capable of receiving, and following the same order. Also the high order bit in the type, is the critical bit, MUST be set accordingly.

The above sub-TLV(s) MAY be included with only Ethernet A-D per-ES routes.

4. Operation

The following figure shows an example of an NVO3 deployment with EVPN.

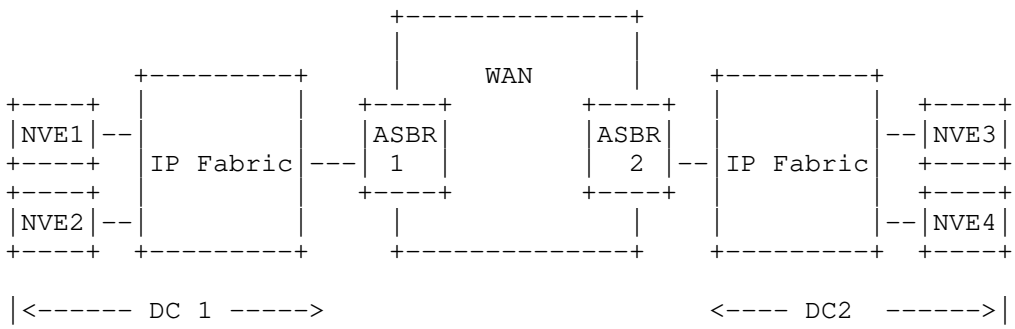


Figure 2: Data Center Interconnect with ASBR

iBGP sessions are established between NVE1, NVE2, ASBR1, possibly via a BGP route-reflector. Similarly, iBGP sessions are established between NVE3, NVE4, ASBR2.

eBGP sessions are established among ASBR1 and ASBR2.

All NVEs and ASBRs are enabled for the EVPN SAFI and exchange EVPN routes. For inter-AS option B, the ASBRs re-advertise these routes with NEXT_HOP attribute set to their IP addresses as per [RFC4271].

NVE1 sets the BGP Encapsulation extended community defined in all EVPN routes advertised. NVE1 sets the BGP Tunnel Encapsulation Attribute Tunnel Type to Geneve tunnel encapsulation, and sets the Tunnel Encapsulation Attribute Tunnel sub-TLV for the Geneve tunnel option types with all the Geneve option types it can transmit and receive.

All other NVE(s) learn what Geneve option types are supported by NVE1 through the EVPN control plane. In the datapath, NVE2, NVE3 and NVE4 only encapsulate overlay packets with the Geneve option TLV(s) that NVE1 is capable of receiving.

A PE advertises the BGP Encapsulation extended community defined in [RFC5512] if it supports any of the encapsulations defined in [EVPN-OVERLAY]. A PE advertises the BGP Tunnel Encapsulation Attribute defined in [TUNNEL-ENCAP] if it supports Geneve encapsulation.

5. Security Considerations

The mechanisms in this document use EVPN control plane as defined in [RFC7432]. Security considerations described in [RFC7432] are equally applicable.

This document uses IP-based tunnel technologies to support data plane transport. Security considerations described in [RFC7432] and in [EVPN-OVERLAY] are equally applicable.

6. IANA Considerations

IANA is requested to allocate the following:

BGP Tunnel Encapsulation Attribute
Tunnel Type:

XX	Geneve Encapsulation

BGP Tunnel Encapsulation Attribute Sub-TLVs a Code point from the range of 192-252 for Geneve tunnel option types sub-TLV.

IANA is requested to assign a new option class from the "Geneve Option Class" registry for the Ethernet option TLV.

Option Class	Description

XXXX	Ethernet option

7. Acknowledgements

The authors wish to thank T. Sridhar, for his input, feedback, and helpful suggestions.

8. References

8.1 Normative References

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

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <http://www.rfc-editor.org/info/rfc7432>.

[RFC8317] Sajassi, et al. "Ethernet-Tree (E-Tree) Support in Ethernet VPN (EVPN) and Provider Backbone Bridging EVPN (PBB-EVPN)", RFC 8317, January 2018, <http://www.rfc-editor.org/info/rfc8317>.

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

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[GENEVE] Gross, et al. "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-05, work in progress, September, 2017.

[DT-ENCAP] Boutros, et al. "NVO3 Encapsulation Considerations", draft-ietf-nvo3-encap-01, work in progress, October, 2017.

[TUNNEL-ENCAP] Rosen et al., "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-07, work in progress, July, 2017.

[EVPN-OVERLAY] Sajassi-Drake et al., "A Network Virtualization Overlay Solution using EVPN", draft-ietf-bess-evpn-overlay-10.txt, work in progress, December, 2017

8.2 Informative References

[NVO3-FRWK] Lasserre et al., "Framework for DC Network Virtualization", RFC 7365, October 2014.

Authors' Addresses

Sami Boutros
VMware, Inc.
Email: boutross@vmware.com

Ali Sajassi
Cisco
Email: sajassi@cisco.com

John Drake
Juniper Networks
Email: jdrake@juniper.net

Jorge Rabadan
Nokia
Email: jorge.rabadan@nokia.com

Sam Aldrin
Google
Email: aldrin.ietf@gmail.com

BESS Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 7, 2020

P. Brissette, Ed.
A. Sajassi
L. Burdet
Cisco Systems
D. Voyer
Bell Canada

November 4, 2019

EVPN Multi-Homing Mechanism for Layer-2 Gateway Protocols
draft-brissette-bess-evpn-l2gw-proto-05

Abstract

The existing EVPN multi-homing load-balancing modes defined are Single-Active and All-Active. Neither of these multi-homing mechanisms are appropriate to support access networks with Layer-2 Gateway protocols such as G.8032, MPLS-TP, STP, etc. These Layer-2 Gateway protocols require a new multi-homing mechanism defined in this draft.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on May 7, 2020.

Copyright Notice

Table of Contents

1. Introduction

Existing EVPN multi-homing mechanisms of Single-Active and All-Active are not sufficient to support access Layer-2 Gateway protocols such as G.8032, MPLS-TP, STP, etc.

These Layer-2 Gateway protocols require that a given flow of a VLAN (represented by {MAC-SA, MAC-DA}) to be only active on one of the PEs in the multi-homing group. This is in contrast with Single-Active redundancy mode where all flows of a VLAN are active on one of the multi-homing PEs and it is also in contrast with All-Active redundancy mode where all L2 flows of a VLAN are active on all PEs in the redundancy group.

This draft defines a new multi-homing mechanism "Single-Flow-Active" which defines that a VLAN can be active on all PEs in the redundancy group but a single given flow of that VLAN can be active on only one of the PEs in the redundancy group. In fact, the carving scheme, performed by the DF(Designated Forwarder) election algorithm for

these L2 Gateway protocols, is not per VLAN but rather for a given VLAN. A selected PE in the redundancy group can be the only Designated Forwarder for a specific L2 flow but the decision is not taken by the PE. The loop-prevention blocking scheme occurs in the access network.

EVPN multi-homing procedures need to be enhanced to support Designated Forwarder election for all traffic (both known unicast and BUM) on a per L2 flow basis. This new multi-homing mechanism also requires new EVPN considerations for aliasing, mass-withdraw, fast-switchover and [EVPN-IRB] as described in the solution section.

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 [RFC2119].

1.2. Terms and Abbreviations

AC: Attachment Circuit

BUM: Broadcast, Unknown unicast, Multicast

DF: Designated Forwarder

GW: Gateway

L2 Flow: A given flow of a VLAN, represented by (MAC-SA, MAC-DA)

L2GW: Layer-2 Gateway

G.8032: Ethernet Ring Protection

MST-AG: Multi-Spanning Tree Access Gateway

REP-AG: Resilient Ethernet Protocol Access Gateway

TCN: Topology Change Notification

2. Solution

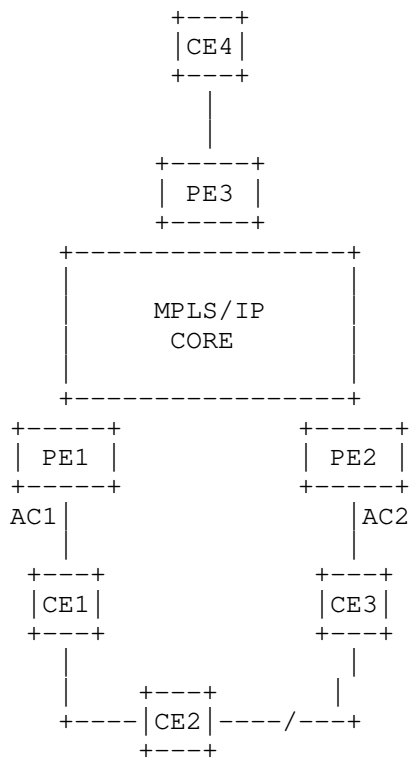


Figure 1: EVPN network with L2 access GW protocols

Figure 1 shows a typical EVPN network with an access network running a L2GW protocol, typically one of the following: G.8032, STP, MPLS-TP, etc. The L2GW protocol usually starts from AC1 (on PE1) up to AC2 (on PE2) in an open "ring" manner. AC1 and AC2 interfaces of PE1 and PE2 are participants in the access protocol.

The L2GW protocol is used for loop avoidance. In above example, the loop is broken on the right side of CE2.

2.1. Single-Flow-Active redundancy mode

PE1 and PE2 are peering PEs in a redundancy group, and sharing a same ESI. In the proposed Single-Flow-Active mode, PE1 and PE2 'Access Gateway' load-balancing mode shares similarities with both Single-Active and All-Active. DF election must not result in blocked ports or portions of the access may become isolated. Additionally, the reachability between CE1/CE2 and CE3 is achieved with the forwarding path through the EVPN MPLS/IP core side. Thus, the ESI-Label

filtering of [RFC7432] is disabled for Single-Flow-Active Ethernet segments.

Finally, PE3 behaves according to EVPN rules for traffic to/from PE1/PE2. Peering PE, selected per L2 flow, is chosen by the L2GW protocol in the access, and is out of EVPN control.

From PE3 point of view, some of the L2 flows coming from PE3 may reach CE3 via PE2 and some of the L2 flows may reach CE1/CE2 via PE1. A specific L2 flow never goes to both peering PEs. Therefore, aliasing cannot be performed by PE3. That node operates in a single-active fashion for each of these L2 flows.

The backup path which is also setup for rapid convergence, is not applicable here. For example, in Figure 1, if a failure happens between CE1 and CE2, L2 flows coming from CE4 behind PE3 destined to CE1 still goes through PE1 and shall not switch to PE2 as a backup path. On PE3, there is no way to know which L2 flow specifically is affected. During the transition time, PE3 may flood until unicast traffic recovers properly.

2.2. Backwards compatibility

2.2.1. The two-ESI solution

As background, an alternative solution which achieves some, but not all, of the requirements exists and is backwards compatible with [RFC7432]:

On the PE1 and PE2,

a. A single-homed (different) non-zero ESI, or zero-ESI, is used for each PE;

b. With no remote Ethernet-Segment routes received matching local ESI, each PE will be designated forwarder for all the local VLANs;

c. Each L2GW PE will send Ethernet AD per-ES and per-EVI routes for its ESI if non-zero; and

d. When the L2GW PEs receive a MAC-Flush notification (STP TCN, G.8032 mac-flush, LDP MAC withdrawal etc.), they send an update of the Ethernet AD per-EVI route with the MAC Mobility extended community defined in Section 6 and a higher sequence number.

While this solution is feasible, it is considered to fall short of the requirements listed in Section 3, namely for all aspects meant to achieve fast-convergence.

2.2.2. RFC7432 Remote PE

A PE which receives an Ethernet AD per ES route with the Single-Flow-Active bit set in the ESI-flags, and which does not support/ understand this bit, SHALL discard the bit and continue operating per [RFC7432] (All-Active). The operator should understand the usage of single-flow-active load-balancing mode else it is highly recommended to use the two-ESI approach as described in section 2.2.1.

The remote PE3 which does not support Single-Flow-Active redundancy mode as described, will ECMP traffic to peering PEs PE1 and PE2 in the example topology above (Figure 1), per [RFC7432], Section 8.4 aliasing and load-balancing rules. PE1 and PE2, which support the Single-Flow-Active redundancy mode MUST setup sub-optimal Layer-2 forwarding and sub-optimal Layer-3 routing towards the PE at which the flow is currently active.

Thus, while PE3 is ECMP (on average) 50% of the traffic to the incorrect PE in [RFC7432] operation, PE1 and PE2 will handle this gracefully in Single-Flow-Active mode and redirect across peering pair of PEs appropriately.

No extra route or information is required for this. The [RFC7432] and [EVPN-IRB] route advertisements are sufficient.

3. Requirements

The EVPN L2GW framework for L2GW protocols in Access-Gateway mode, consists of the following rules:

- Peering PEs MUST share the same ESI.
- The Ethernet-Segment DF election MUST NOT be performed and forwarding state MUST be dictated by the L2GW protocol. In Access Gateway mode, both PEs are usually in forwarding state. In fact, access protocol is responsible for operationally setting the forwarding state for each VLAN.
- Split-horizon filtering is NOT needed because L2GW protocol ensures there will never be loop in the access network. The forwarding between peering PEs MUST also be preserved. In figure 1, CE1/CE2 device may need reachability with CE3 device. ESI-filtering capability MUST be disabled. PE MUST NOT advertise

corresponding ESI-label to other PEs in the redundancy group, or apply it if it is received.

- ESI-label BGP-extcomm MUST support a new multi-homing mode named "Single-Flow-Active" corresponding to the single-active behaviour of [RFC7432], applied per flow.

- Upon receiving ESI-label BGP-Extcomm with the single-flow-active load-balancing mode, remote PE MUST:

  * Disable ESI-Label processing

  * Disable aliasing (at Layer-2 and Layer-3 [EVPN-IRB])

- The Ethernet-Segment procedures in the EVPN core such as Ethernet AD per-ES and per Ethernet AD per-EVI routes advertisement/ withdraw, as well as MAC and MAC+IP advertisement, remains as explained in [RFC7432] and [EVPN-IRB].

- For fast-convergence, remote PE3 MAY set up two distinct backup paths on a per-flow basis:

  * { PE1 active, PE2 backup }

  * { PE2 active, PE1 backup }

  The backup paths so created, operate as in [RFC7432] section 8.4 where the backup PE of the redundancy group MAY immediately be selected for forwarding upon detection of a specific subset of failures: Ethernet AD per-ES route withdraw, Active PE loss of reachability (via IGP detection). An Ethernet AD per-EVI withdraw MUST NOT result in automatic switching to the backup PE as only a subset of the hosts may be changing reachability to the Backup PE, and the remote cannot determine which.

- MAC mobility procedures SHALL have precedence in Single-Flow-Active for tracking host reachability over backup path procedure.

4. Handling of Topology Change Notification (TCN)

In order to address rapid Layer-2 convergence requirement, topology change notification received from the L2GW protocols must be sent across the EVPN network to perform the equivalent of legacy L2VPN remote MAC flush.

The generation of TCN is done differently based on the access protocol. In the case of STP (REP-AG) and G.8032, TCN gets generated in both directions and thus both of the dual-homing PEs receive it.

However, with STP (MST-AG), TCN gets generated only in one direction and thus only a single PE can receive it. That TCN is propagated to the other peering PE for local MAC flushing, and relaying back into the access.

In fact, PEs have no direct visibility on failures happening in the access network neither on the impact of those failures over the connectivity between CE devices. Hence, both peering PEs require to perform a local MAC flush on corresponding interfaces.

There are two options to relay the access protocol's TCN to the peering PE: in-band or out-of-band messaging. The first method is better for rapid convergence, and requires a dedicated channel between peering PEs. An EVPN-VPWS connection MAY be dedicated for that purpose, connecting the Untagged ACs of both PEs. The latter choice relies on a new MAC flush extended community in the Ethernet Auto-discovery per EVI route, defined below. It is a slower method but has the advantage of avoid the usage of a dedicated channel between peering PEs.

Peering PE, upon receiving TCN from access, MUST:

- As per legacy VPLS, perform a local MAC flush on the access-facing interfaces. An ARP probe is also sent for all hosts previously locally-attached.

- Advertise per EVI/EAD route along with a new MAC-flush BGP Extended Community in order to perform a remote MAC flush and steer L2 traffic to proper peering PE. The sequence number is incremented by one as a flushing indication to remote PEs.

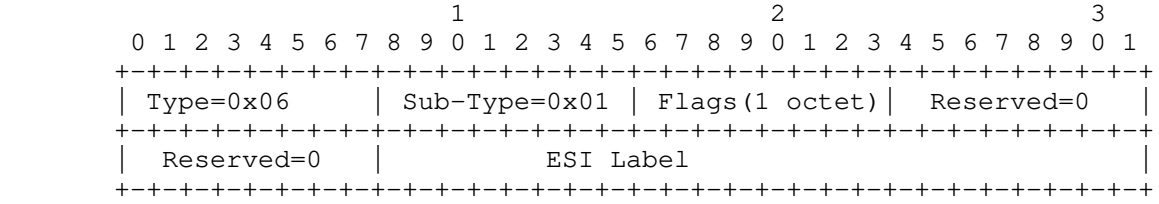
- Ensure MAC and MAC/IP route re-advertisement, with incremented sequence number when host reachability is NOT moving to peering PE. This is to ensure a re-advertisement of current MAC and MAC/IP which may have been flushed remotely upon MAC Flush extcomm reception. In theory, it should happen automatically since peering PE, receiving TCN from the access, performs local MAC flush on corresponding interface and will re-learn that local MAC or MAC/IP at ARP probe reply.

- Where an access protocol relies on TCN BPDU propagation to all participant nodes, a dedicated EVPN-VPWS connection MAY be used as an in-band channel to relay TCN between peering PEs. That connection may be auto-generated or can simply be directly configured by user.

5. ESI-label Extended Community Extension

In order to support the new EVPN load-balancing mode (single-flow-active), the ESI-label extended community is updated.

The 1 octet flag field, part of the ESI Label extended community, is modified as follows:



Low-order bit: [7:0]
[2:0]- 000 = all-active,
001 = single-active,
010 = single-flow-active,
others = unassigned
[7:3]- Reserved

Figure 2: ESI Label extended community

6. EVPN MAC-Flush Extended Community

The MAC mobility BGP Extended community, is required for the TCN procedures and MAC-Flushing. The well-known MAC-Flush procedure from [RFC7623] is borrowed, only for Ethernet AD per-EVI routes.

In this Single-Flow-Active mode, the MAC-Flush Extended Community is advertised along with Ethernet AD per EVI routes upon reception of TCN from the access. When this extended community is used, it indicates, to all remote PEs that all MAC addresses associated with that EVI/ESI are "flushed" i.e. unresolved. They remain unresolved until remote PE receives a route update / withdraw for those MAC addresses; the MAC may be re-advertised by the same PE, or by another, in the same ESI.

The sequence number used is of local significance from the originating PE, and is not used for comparison between peering PEs. Rather, it is used to signal via BGP successive MAC Flush requests from a given PE.

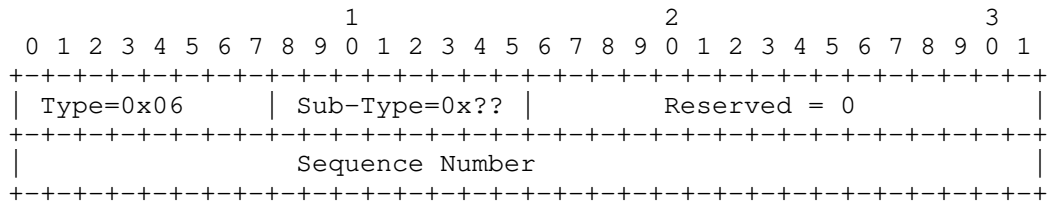


Figure 3: MAC-Flush Extended Community

7. EVPN Inter-subnet Forwarding

EVPN Inter-subnet forwarding procedures in [EVPN-IRB] works with the current proposal and does not require any extension. Host routes continue to be installed at PE3 with a single remote nexthop, no aliasing.

However, leveraging the same-ESI on both L2GW PEs enables ARP/ND synchronization procedures which are defined for All-Active redundancy in [EVPN-IRB]. In steady-state, on PE2 where a host is not locally-reachable the routing table will reflect PE1 as the destination. However, with ARP/ND synchronization based on a common ESI, the ARP/ND cache may be pre-populated with the local AC as destination for the host, should an AC failure occur on PE1. This achieves fast-convergence.

When a hosts moves to PE2 from the PE1 L2GW peer, the MAC mobility sequence number is incremented to signal to remote peers that a 'move' has occurred and the routing tables must be updated to PE2. This is required when an Access Protocol is running where the loop is broken between two CEs in the access and the L2GWs, and the host is no longer reachable from the PE1-side but now from the PE2-side of the access network.

8. Conclusion

EVPN style="symbols"Multi-Homing Mechanism for Layer-2 gateway Protocols solves a true problem due to the wide legacy deployment of these access L2GW protocols in Service Provider networks. The current draft has the main advantage to be fully compliant with [RFC7432] and [EVPN-IRB].

9. Security Considerations

The same Security Considerations described in [RFC7432] and [EVPN-IRB] remain valid for this document.

10. Acknowledgements

Authors would like to thank Thierry Couture for valuable review and inputs with respect to access protocol deployments related to procedures proposed in this document.

11. IANA Considerations

A new allocation of Extended Community Sub-Type for EVPN is required to support the new EVPN MAC flush mechanism..

12. References

12.1. Normative References

[EVPN-IRB] Sajassi, A., "Integrated Routing and Bridging in EVPN", 2019.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

12.2. Informative References

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

Authors' Addresses

Patrice Brissette (editor)
Cisco Systems
Ottawa, ON
Canada

Email: pbrisset@cisco.com

Ali Sajassi
Cisco Systems
USA

Email: sajassi@cisco.com

Luc Andre Burdet
Cisco Systems
Ottawa, ON
Canada

Email: lburdet@cisco.com

Daniel Voyer
Bell Canada
Montreal, QC
Canada

Email: daniel.voyer@bell.ca

INTERNET-DRAFT
Intended Status: Proposed Standard

Patrice Brissette
Samir Thoria
Ali Sajassi
Cisco Systems

Expires: September 1, 2018

February 28, 2018

EVPN multi-homing port-active load-balancing
draft-brissette-bess-evpn-mh-pa-01

Abstract

The Multi-Chassis Link Aggregation Group (MC-LAG) technology enables the establishment of a logical port-channel connection with a redundant group of independent nodes. The purpose of multi-chassis LAG is to provide a solution to achieve higher network availability, while providing different modes of sharing/balancing of traffic. EVPN standard defines EVPN based MC-LAG with single-active and all-active multi-homing load-balancing mode. The current draft expands on existing redundancy mechanisms supported by EVPN and introduces support of port-active load-balancing mode. In the current draft, port-active load-balancing mode is also referred to as per interface active/standby.

Table of Contents

1 Introduction

EVPN, as per [RFC7432], provides all-active per flow load balancing for multi-homing. It also defines single-active with service carving mode, where one of the PEs, in redundancy relationship, is active per service.

While these two multi-homing scenarios are most widely utilized in data center and service provider access networks, there are scenarios where active-standby per interface multi-homing redundancy is useful and required. Main consideration for this mode of redundancy is the determinism of traffic forwarding through specific interface rather than statistical per flow load balancing across multiple PEs providing multi-homing. The determinism provided by active-standby per interface is also required for certain QOS features to work. While using this mode, customers also expect minimized convergence during failures. A new term of load-balancing mode "port-active load-balancing" is then defined.

This draft describes how that new redundancy mode can be supported via EVPN.

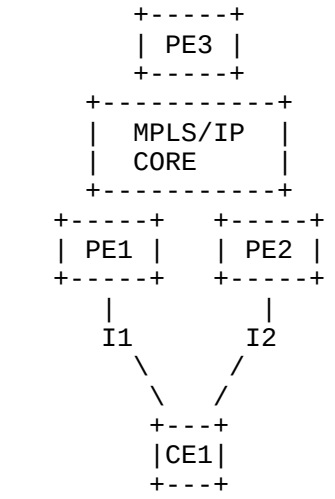


Figure 1. MC-LAG topology

Figure 1 shows a MC-LAG multi-homing topology where PE1 and PE2 are part of the same redundancy group providing multi-homing to CE1 via interfaces I1 and I2. Interfaces I1 and I2 are Bundle-Ethernet interfaces running LACP protocol. The core, shown as IP or MPLS enabled, provides wide range of L2 and L3 services. MC-LAG multi-homing functionality is decoupled from those services in the core and

it focuses on providing multi-homing to CE. With per-port active/standby redundancy, only one of the two interface I1 or I2 would be in forwarding, the other interface will be in standby. This also implies that all services on the active interface are in active mode and all services on the standby interface operate in standby mode. When EVPN is used to provide MC-LAG functionality, we refer to it as EVLAG in this draft.

1.1 Terminology

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

2. Multi-Chassis Ethernet Bundles

When a CE is multi-homed to a set of PE nodes using the [802.1AX] Link Aggregation Control Protocol (LACP), the PEs must act as if they were a single LACP speaker for the Ethernet links to form a bundle, and operate as a Link Aggregation Group (LAG). To achieve this, the PEs connected to the same multi-homed CE must synchronize LACP configuration and operational data among them. ICCP-based protocol has been used for that purpose since a long while. EVLAG simplifies greatly that solution. Along with the simplification comes few assumptions:

- Links in the Ethernet Bundle MUST operate in all-active load-balancing mode

- Same LACP parameters MUST be configured on peering PEs such as system id, port priority, etc.

Any discrepancies from this list is left for future study. Furthermore, mis-configuration and mis-wiring detection across peering PEs are also left for further study.

3. Port-active load-balancing procedure

Following steps describe the proposed procedure with EVLAG to support port-active load-balancing mode:

1- ESI MUST be assigned per access interface as described in [RFC7432], which may be auto derived or manually assigned. Access interface MAY be a Layer-2 or Layer3 interface.

2- Ethernet-Segment MUST be configured in port-active load-balancing mode on peering PEs for specific interface

3- Peering PEs MAY exchange only Ethernet-Segment route (Route Type-4)

4- PEs in the redundancy group leverages DF election defined in [draft-ietf-bess-evpn-df-election] to determine which PE keeps the port in active mode and which one(s) keep it in standby mode. While the DF election defined in [draft-ietf-bess-evpn-df-election] is per <ES, VLAN> granularity, for port-active mode of multi-homing, the DF election is done per <ES>. The details of this algorithm are described in Section 4.

5- DF router MUST keep corresponding access interface in up and forwarding active state for that Ethernet-Segment

6- Non-DF routers MUST bring and keep peering access interface attached to it in operational down state. If the interface is running LACP protocol, then the non-DF PE MAY also set the LACP state to OOS (Out of Sync) as opposed to interface state down. This allows for better convergence on standby to active transition.

4. Algorithm to elect per port-active PE

The default mode of Designated Forwarder Election algorithm remains as per [RFC7432] at the granularity of <ES>.

However, Highest Random Weight (HRW) algorithm defined in [draft-ietf-bess-evpn-df-election] is leveraged, and modified to operate at the granularity of <ES> rather than per <ES, VLAN>.

Let Active(ESI) denote the PE that will be the active PE for port with Ethernet segment identifier - ESI. The other PEs in the redundancy group will be standby PE(s) for the same port (ES). Ai is the address of the PEi and weight() is a pseudorandom function of ESi and Ai, Wrand() function defined in [draft-ietf-bess-evpn-df-election] is used as the Weight() function.

Active(ESI) = PEi: if Weight(ESI, Ai) >= Weight(ESI, Aj), for all j, 0 <= I,j <= Number of PEs in the redundancy group. In case of a tie, choose the PE whose IP address is numerically the least.

5. Port-active over Integrated Routing-Bridging Interface

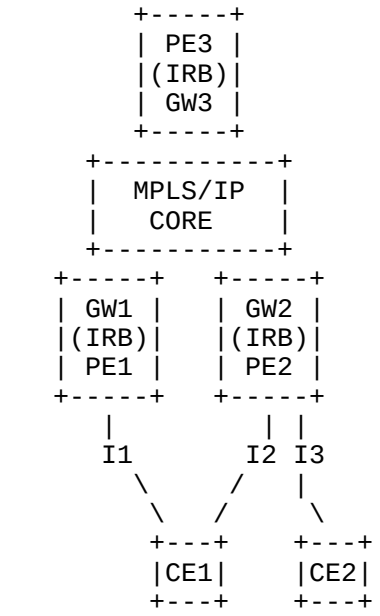


Figure 2. EVPN-IRB Port-active load-balancing

Figure 2 shows a simple network where EVPN-IRB is used for inter-subnet connectivity. IRB interfaces on PE1 and PE2 are configured in anycast gateway (same MAC, same IP). CE1 device is multi-homed to both PE1 and PE2. The Ethernet-segment load-balancing mode, of the connected CE1 to peering PEs, can be of any type e.g. all-active, single-active or port-active. CE2 device is connected to a single PE (PE2). It operates as single-homed device via an orphan port I3. Finally, port-active load-balancing is apply to IRB interface on peering PEs (PE1 and PE2). Manual Ethernet-Segment Identifier is assigned per IRB interface. ESI auto-generation is also possible based on the IRB anycast IP address.

DF election is performed between peering PE over IRB interface (per ESI/EVI). Designed forwarder (DF) IRB interface remains in up state. Non-designated forwarder (NDF) IRB interface goes down. Furthermore, if all access interfaces connected to an IRB interface are down state (failure or admin) OR in blocked forward state(NDF), IRB interface is brought down. For example, interface I3 fails at the same time than interface I2 (in single-active load-balancing mode) is in blocked forwarding state.

In the example where IRB on PE2 is NDF, all L3 traffic coming from

PE3 is going via PE1. An IRB interface in down state doesn’t attract traffic from core side. CE2 device reachability is done via an L2 subnet stretch between PE1 and PE2. Therefore L3 traffic coming from PE3 destinated to CE2 goes via GW1 first, then via an L2 connection to PE2 and finally via interface I3 to CE2 device.

There are many reasons of configuring port-active load-balancing mode over IRB interface:
- Ease replacement of legacy technology such VRRP / HSRP

- Better scalability than legacy protocols

- Traffic predictability

- Optimal routing and entirely independent of load-balancing mode configured on any access interfaces

6. Convergence considerations

To improve the convergence, upon failure and recovery, when port-active load-balancing mode is used, some advanced synchronization between peering PEs may be required. Port-active is challenging in a sense that the "standby" port is in down state. It takes some time to bring a "standby" port in up-state and settle the network. For IRB and L3 services, ARP / MLD cache may be synchronized. Moreover, associated VRF tables may also be synchronized. For L2 services, MAC table synchronization may be considered. Finally, using bundle-Ethernet interface, where LACP is running, is usually a smart thing since it provides the ability to set the "standby" port in "out-of-sync" state aka "warm-standby".

6. Applicability

A common deployment is to provide L2 or L3 service on the PEs providing multi-homing. The services could be any L2 EVPN such as EVPN VPWS, EVPN [RFC7432], etc. L3 service could be in VPN context [RFC4364] or in global routing context. When a PE provides first hop routing, EVPN IRB could also be deployed on the PEs. The mechanism defined in this draft is used between the PEs providing the L2 or L3 service, when the requirement is to use per port active.

A possible alternate solution is the one described in this draft is MC-LAG with ICCP [RFC7275] active-standby redundancy. However, ICCP requires LDP to be enabled as a transport of ICCP messages. There are many scenarios where LDP is not required e.g. deployments with VXLAN or SRv6. The solution defined in this draft with EVPN does not mandate the need to use LDP or ICCP and is independent of the overlay encapsulation.

7. Overall Advantages

There are many advantages in EVLAG to support port-active load-balancing mode. Here is a non-exhaustive list:

- Open standards based per interface single-active redundancy mechanism that eliminates the need to run ICCP and LDP.

- Agnostic of underlay technology (MPLS, VXLAN, SRv6) and associated services (L2, L3, Bridging, E-LINE, etc).

- Provides a way to enable deterministic QOS over MC-LAG attachment circuits

- Fully compliant with RFC-7432, does not require any new protocol enhancement to existing EVPN RFCs.

- Can leverage various DF election algorithms e.g. modulo, HRW, etc.

- Replaces legacy MC-LAG ICCP-based solution, and offers following additional benefits:

    - Efficiently supports 1+N redundancy mode (with EVPN using BGP RR) where as ICCP requires full mesh of LDP sessions among PEs in redundancy group

    - Fast convergence with mass-withdraw is possible with EVPN, no equivalent in ICCP

- Customers want per interface single-active redundancy, but don’t want to enable LDP (e.g. they may be running VXLAN or SRv6 in the network). Currently there is no alternative to this.

8 Security Considerations

The same Security Considerations described in [RFC7432] are valid for this document.

9 IANA Considerations

There are no new IANA considerations in this document.

10 References

10.1 Normative References

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7275] Martini, L., Salam, S., Sajassi, A., Bocci, M., Matsushima, S., and T. Nadeau, "Inter-Chassis Communication Protocol for Layer 2 Virtual Private Network (L2VPN) Provider Edge (PE) Redundancy", RFC 7275, DOI 10.17487/RFC7275, June 2014, <https://www.rfc-editor.org/info/rfc7275>.

10.2 Informative References

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

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

Authors' Addresses

Patrice Brissette
Cisco Systems
EMail: pbrisset@cisco.com

Samir Thoria
Cisco Systems

EMail: sthoria@cisco.com

Ali Sajassi
Cisco Systems
EMail: sajassi@cisco.com

BESS Working Group
INTERNET-DRAFT
Intended Status: Proposed Standard

Patrice Brissette
Ali Sajassi
Cisco Systems

Bin Wen
Comcast

Edward Leyton
Verizon Wireless

Jorge Rabadan
Nokia

Expires: May 3, 2020

October 31, 2019

EVPN multi-homing port-active load-balancing
draft-brissette-bess-evpn-mh-pa-04

Abstract

The Multi-Chassis Link Aggregation Group (MC-LAG) technology enables the establishment of a logical link-aggregation connection with a redundant group of independent nodes. The purpose of multi-chassis LAG is to provide a solution to achieve higher network availability, while providing different modes of sharing/balancing of traffic. EVPN standard defines EVPN based MC-LAG with single-active and all-active multi-homing load-balancing mode. The current draft expands on existing redundancy mechanisms supported by EVPN and introduces support of port-active load-balancing mode. In the current document, port-active load-balancing mode is also referred to as per interface active/standby.

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

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

Table of Contents

1 Introduction

EVPN, as per [RFC7432], provides all-active per flow load balancing for multi-homing. It also defines single-active with service carving mode, where one of the PEs, in redundancy relationship, is active per service.

While these two multi-homing scenarios are most widely utilized in data center and service provider access networks, there are scenarios where active-standby per interface multi-homing redundancy is useful and required. The main consideration for this mode of redundancy is the determinism of traffic forwarding through a specific interface rather than statistical per flow load balancing across multiple PEs providing multi-homing. The determinism provided by active-standby per interface is also required for certain QOS features to work. While using this mode, customers also expect minimized convergence during failures. A new term of load-balancing mode "port-active load-balancing" is then defined.

This draft describes how that new redundancy mode can be supported via EVPN.

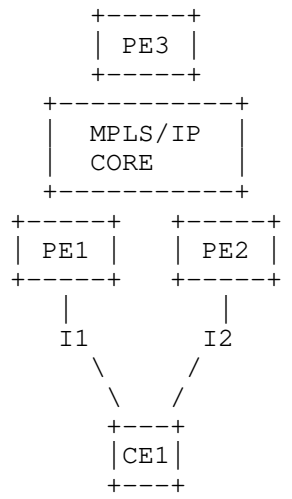


Figure 1. MC-LAG topology

Figure 1 shows a MC-LAG multi-homing topology where PE1 and PE2 are part of the same redundancy group providing multi-homing to CE1 via interfaces I1 and I2. Interfaces I1 and I2 are Bundle-Ethernet interfaces running LACP protocol. The core, shown as IP or MPLS enabled, provides wide range of L2 and L3 services. MC-LAG multi-homing functionality is decoupled from those services in the core and

it focuses on providing multi-homing to CE. With per-port active/standby redundancy, only one of the two interface I1 or I2 would be in forwarding, the other interface will be in standby. This also implies that all services on the active interface are in active mode and all services on the standby interface operate in standby mode.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Multi-Chassis Ethernet Bundles

When a CE is multi-homed to a set of PE nodes using the [802.1AX] Link Aggregation Control Protocol (LACP), the PEs must act as if they were a single LACP speaker for the Ethernet links to form a bundle, and operate as a Link Aggregation Group (LAG). To achieve this, the PEs connected to the same multi-homed CE must synchronize LACP configuration and operational data among them. InterChassis Communicated-based Protocol (ICCP) has been used for that purpose. EVPN LAG simplifies greatly that solution. Along with the simplification comes few assumptions:

- CE device connected to Multi-homing PEs may has a single LAG with all its active links i.e. Links in the Ethernet Bundle operate in all-active load-balancing mode.

- Same LACP parameters MUST be configured on peering PEs such as system id, port priority and port key.

Any discrepancies from this list is left for future study. Furthermore, mis-configuration and mis-wiring detection across peering PEs are also left for further study.

3. Port-active load-balancing procedure

Following steps describe the proposed procedure with EVPN LAG to support port-active load-balancing mode:

1- The Ethernet-Segment Identifier (ESI) MUST be assigned per access interface as described in [RFC7432], which may be auto derived or manually assigned. Access interface MAY be a Layer-2 or Layer3 interface. The usage of ESI over L3 interfce is newly described in this document.

2- Ethernet-Segment MUST be configured in port-active load-balancing mode on peering PEs for specific access interface

3- Peering PEs MAY exchange only Ethernet-Segment route (Route Type-4) when ESI is configured on a Layer3 interface.

4- PEs in the redundancy group leverage the DF election defined in [RFC8584] to determine which PE keeps the port in active mode and which one(s) keep it in standby mode. While the DF election defined in [RFC8584] is per <ES, Ethernet Tag> granularity, for port-active mode of multi-homing, the DF election is done per <ES>. The details of this algorithm are described in Section 4.

5- DF router MUST keep corresponding access interface in up and forwarding active state for that Ethernet-Segment

6- Non-DF routers MAY bring and keep peering access interface attached to it in operational down state. If the interface is running LACP protocol, then the non-DF PE MAY also set the LACP state to OOS (Out of Sync) as opposed to interface state down. This allows for better convergence on standby to active transition.

7- For EVPN-VPWS service, the usage of primary/backup bits of EVPN Layer2 attributes extended community [RFC8214] is highly recommended to achieve better convergence.

4. Algorithm to elect per port-active PE

The ES routes, running in port-active load-balancing mode, are advertised with a new capability in the DF Election Extended Community as defined in [RFC8584]. Moreover, the ES associated to the port leverages existing procedure of single-active, and signals single-active bit along with Ethernet-AD per-ES route. Finally, as in RFC7432, the ESI-label based split-horizon procedures should be used to avoid transient echo'ed packets when L2 circuits are involved.

4.1 Capability Flag

[RFC8584] defines a DF Election extended community, and a Bitmap field to encode "capabilities" to use with the DF election algorithm in the DF algorithm field. Bitmap (2 octets) is extended by the following value:

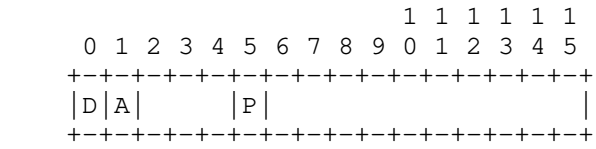


Figure 2 - Amended Bitmap field in the DF Election Extended Community

- Bit 0: 'Don't Preempt' bit, as explained in [PREF-DF].

- Bit 1: AC-Influenced DF Election, as explained in [RFC8584].

- Bit 5: (corresponds to Bit 25 of the DF Election Extended Community and it is defined by this document): P bit or 'Port Mode' bit (P hereafter), determines that the DF-Algorithm should be modified to consider the port only and not the Ethernet Tags.

4.2 Modulo-based Designated Forwarder Algorithm

The default DF Election algorithm, or modulus-based algorithm as in [RFC7432] and updated by [RFC8584], is used here, at the granularity of <ES> only. Given the fact, ES-Import RT community inherits from ESI only byte 1-7, many deployments differentiate ESI within these bytes only. For Modulo calculation, bytes [3-7] are used to determine the designated forwarder using Modulo-based DF assignment.

4.3 HRW Algorithm

Highest Random Weight (HRW) algorithm defined in [RFC8584] MAY also be used and signaled, and modified to operate at the granularity of <ES> rather than per <ES, VLAN>.

[RFC8584] describes computing a 32 bit CRC over the concatenation of Ethernet Tag and ESI. For port-active load-balancing mode, the Ethernet Tag is simply removed from the CRC computation.

4.4 Preferred-DF Algorithm

When the new capability 'Port-Mode' is signaled, the algorithm is modified to consider the port only and not any associated Ethernet Tags. Furthermore, the "port-based" capability MUST be compatible with the 'DP' capability (for non-revertive). The AC-DF bit MUST be set to zero. When an AC (sub-interface) goes down, it does not influence the DF election.

5. Convergence considerations

To improve the convergence, upon failure and recovery, when port-active load-balancing mode is used, some advanced synchronization between peering PEs may be required. Port-active is challenging in a sense that the "standby" port is in down state. It takes some time to bring a "standby" port in up-state and settle the network. For IRB and L3 services, ARP / ND cache may be synchronized. Moreover,

associated VRF tables may also be synchronized. For L2 services, MAC table synchronization may be considered.

Finally, for Bundle-Ethernet interface where LACP is running the ability to set the "standby" port in "out-of-sync" state aka "warm-standby" can be leveraged.

6. Applicability

A common deployment is to provide L2 or L3 service on the PEs providing multi-homing. The services could be any L2 EVPN such as EVPN VPWS, EVPN [RFC7432], etc. L3 service could be in VPN context [RFC4364] or in global routing context. When a PE provides first hop routing, EVPN IRB could also be deployed on the PEs. The mechanism defined in this draft is used between the PEs providing the L2 and/or L3 service, when the requirement is to use per port active.

A possible alternate solution is the one described in this draft is MC-LAG with ICCP [RFC7275] active-standby redundancy. However, ICCP requires LDP to be enabled as a transport of ICCP messages. There are many scenarios where LDP is not required e.g. deployments with VXLAN or SRv6. The solution defined in this draft with EVPN does not mandate the need to use LDP or ICCP and is independent of the underlay encapsulation.

7. Overall Advantages

The use of port-active multi-homing brings the following benefits to EVPN networks:

- Open standards based per interface single-active redundancy mechanism that eliminates the need to run ICCP and LDP.

- Agnostic of underlay technology (MPLS, VXLAN, SRv6) and associated services (L2, L3, Bridging, E-LINE, etc).

- Provides a way to enable deterministic QOS over MC-LAG attachment circuits

- Fully compliant with [RFC7432], does not require any new protocol enhancement to existing EVPN RFCs.

- Can leverage various DF election algorithms e.g. modulo, HRW, etc.

- Replaces legacy MC-LAG ICCP-based solution, and offers following additional benefits:

    - Efficiently supports 1+N redundancy mode (with EVPN using BGP

RR) where as ICCP requires full mesh of LDP sessions among PEs in redundancy group

- Fast convergence with mass-withdraw is possible with EVPN, no equivalent in ICCP

- Customers want per interface single-active redundancy, but don't want to enable LDP (e.g. they may be running VXLAN or SRv6 in the network). Currently there is no alternative to this.

8 Security Considerations

The same Security Considerations described in [RFC7432] are valid for this document.

9 IANA Considerations

This document solicits the allocation of the following values:

- Bit 5 in the [RFC8584] DF Election Capabilities registry, with name "P"(port mode load-balancing) Capability" for port-active ES.

10 References

10.1 Normative References

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

[RFC8584] Rabadan, J., Ed., Mohanty, S., Ed., Sajassi, A., Drake, J., Nagaraj, K., and S. Sathappan, "Framework for Ethernet VPN Designated Forwarder Election Extensibility", RFC 8584, DOI 10.17487/RFC8584, April 2019, <https://www.rfc-editor.org/info/rfc8584>.

10.2 Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate

Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

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

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC7275] Martini, L., Salam, S., Sajassi, A., Bocci, M., Matsushima, S., and T. Nadeau, "Inter-Chassis Communication Protocol for Layer 2 Virtual Private Network (L2VPN) Provider Edge (PE) Redundancy", RFC 7275, DOI 10.17487/RFC7275, June 2014, <https://www.rfc-editor.org/info/rfc7275>.

[PREF-DF] Rabadan et al. "Preference-based EVPN DF Election", draft-ietf-bess-evpn-pref-df, work-in-progress, June, 2019.

Authors' Addresses

Patrice Brissette
Cisco Systems
EMail: pbrisset@cisco.com

Ali Sajassi
Cisco Systems
EMail: sajassi@cisco.com

Luc Andre Burdet
Cisco Systems
EMail: lburdet@cisco.com

Samir Thoria
Cisco Systems
EMail: sthoria@cisco.com

Jorge Rabadan
Nokia
Email: jorge.rabadan@nokia.com

Bin Wen

Comcast
Email: Bin_Wen@comcast.com

Edward Leyton
Verizon
Email: edward.leyton@verizonwireless.com

INTERNET-DRAFT
Intended status: Proposed Standard

V. Govindan
M. Mudigonda
A. Sajassi
Cisco Systems
G. Mirsky
ZTE
D. Eastlake
Huawei

Expires: November 24, 2018
May 25, 2018

Fault Management for EVPN networks
draft-gmsm-bess-evpn-bfd-01

Abstract

This document specifies a proactive, in-band network OAM mechanism to detect loss of continuity and miss-connection faults that affect unicast and multi-destination paths, used by Broadcast, unknown Unicast and Multicast traffic, in an EVPN network. The mechanisms proposed in the draft use the widely adopted Bidirectional Forwarding Detection (BFD) protocol.

Status of this Memo

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

Distribution of this document is unlimited. Comments should be sent to the authors or the BESSq working group mailing list: bess@ietf.org.

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/1id-abstracts.html. The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Table of Contents

1. Introduction

[I-D.eastlake-bess-evpn-oam-req-frmwk] and [I-D.ooamdt-rtgwg-ooam-requirement] outline the OAM requirements of Ethernet VPN networks [RFC7432]. This document proposes mechanisms for proactive fault detection at the network (overlay) OAM layer of EVPN. EVPN fault detection mechanisms need to consider unicast traffic separately from Broadcast, unknown Unicast, and Multicsst (BUM) traffic since they map to different FECs in EVPN, hence this document proposes different fault detection mechanisms to suit each type using the principles of [RFC5880], [RFC5884] and Point-to-multipoint BFD [I-D.ietf-bfd-multipoint] and [I-D.ietf-bfd-multipoint-active-tail]. Packet loss and packet delay measurement are out of scope for this document.

1.1 Terminology

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

The following acronyms are used in this document.

BUM - Broadcast, Unknown Unicast, and Multicast

CC - Continuity Check

CV - Connectivity Verification

FEC - Forwarding Equivalency Class

GAL - Generic Associated Channel Label

LSM - Label Switched Multicast (P2MP)

LSP - Label Switched Path

MP2P - Multi-Point to Point

OAM - Operations Administration, and Maintenance

P2MP - Point to Multi-Point (LSM)

PE - Provider Edge

PHP - Penultimate Hop Popping

2. Scope of this Document

This document specifies proactive fault detection for EVPN [RFC7432] using BFD mechanisms for:

- Unicast traffic.
- BUM traffic using Multi-point-to-Point (MP2P) tunnels (ingress replication).
- BUM traffic using Point-to-Multipoint (P2MP) tunnels (LSM).

This document does not discuss BFD mechanisms for:

- EVPN variants like PBB-EVPN [RFC7623]. This will be addressed in future versions.
- Integrated Routing and Bridging (IRB) solution based on EVPN [I-D.ietf-bess-evpn-inter-subnet-forwarding]. This will be addressed in future versions.
- EVPN using other encapsulations like VxLAN, NVGRE and MPLS over GRE [RFC8365].
- BUM traffic using MP2MP tunnels will also be addressed in a future version of this document.

This specification describes procedures only for BFD asynchronous mode. BFD demand mode is outside the scope of this specification. Further, the use of the Echo function is outside the scope of this specification.

3. Motivation for Running BFD at the EVPN Network Layer

The choice of running BFD at the network layer of the OAM model for EVPN [I-D.eastlake-bess-evpn-oam-req-frmwk] and [I-D.ooamdt-rtgwg-ooam-requirement] was made after considering the following:

- In addition to detecting link failures in the EVPN network, BFD sessions at the network layer can be used to monitor the successful programming of labels used for setting up MP2P and P2MP EVPN tunnels transporting Unicast and BUM traffic. The scope of reachability detection covers the ingress and the egress EVPN PE nodes and the network connecting them.
- Monitoring a representative set of path(s) or a particular path among the multiple paths available between two EVPN PE nodes could be done by exercising the entropy labels when they are used.

However paths that cannot be realized by entropy variations cannot be monitored. Fault monitoring requirements outlined by [I-D.eastlake-bess-evpn-oam-req-frmwk] are addressed by the mechanisms proposed by this draft.

Successful establishment and maintenance of BFD sessions between EVPN PE nodes does not fully guarantee that the EVPN service is functioning. For example, an egress EVPN-PE can understand the EVPN label but could switch data to incorrect interface. However, once BFD sessions in the EVPN Network Layer reach UP state, it does provide additional confidence that data transported using those tunnels will reach the expected egress node. When the BFD session in EVPN overlay goes down that can be used as an indication of a Loss-of-Connectivity defect in the EVPN underlay that would cause EVPN service failure.

4. Fault Detection of Unicast Traffic

The mechanisms specified in BFD for MPLS LSPs [RFC5884] [RFC7726] can be applied to bootstrap and maintain BFD sessions for unicast EVPN traffic. The discriminators required for de-multiplexing the BFD sessions MUST be exchanged using EVPN LSP ping specifying the Unicast EVPN FEC [I-D.jain-bess-evpn-lsp-ping] before establishing the BFD session. This is needed since the MPLS label stack does not contain enough information to disambiguate the sender of the packet.

The usage of MPLS entropy labels takes care of the requirement to monitor various paths of the multi-path server layer network [RFC6790]. Each unique realizable path between the participating PE routers MAY be monitored separately when entropy labels are used. The multi-path connectivity between two PE routers MUST be tracked by at least one representative BFD session, but in that case the granularity of fault-detection would be coarser. The PE node receiving the EVPN LSP ping MUST allocate BFD discriminators using the procedures defined in [RFC7726]. Once the BFD session for the EVPN label is UP, the ends of the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first brings down the session as specified in [RFC5884].

5. Fault Detection for BUM Traffic

5.1 Ingress Replication

Ingress replication uses separate MP2P tunnels for transporting BUM traffic from the ingress PE (head) to a set of one or more egress PEs (tails). The fault detection mechanism specified by this document takes advantage of the fact that a unique copy is made by the head for each tail. Another key aspect to be considered in EVPN is the advertisement of the inclusive multicast route. The BUM traffic flows from a head node to a particular tail only after the head receives the inclusive multicast route containing the BUM EVPN label (downstream allocated) corresponding to the MP2P tunnel.

The head-end PE performing ingress replication MUST initiate an EVPN LSP ping using the inclusive multicast FEC [I-D.jain-bess-evpn-lsp-ping] upon receiving an inclusive multicast route from a tail to bootstrap the BFD session. There MAY exist multiple BFD sessions between a head PE an individual tail due to the usage of entropy labels [RFC6790] for an inclusive multicast FEC. The PE node receiving the EVPN LSP ping MUST allocate BFD discriminators using the procedures defined in [RFC7726]. Once the BFD session for the EVPN label is UP, the ends of the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first brings down the session as specified in [RFC5884].

5.2 Label Switched Multicast

Fault detection for BUM traffic distributed by a Label Switched Multicst (LSM) using a P2MP tunnel is done with active tail multipoint BFD in the reliable head notification scenario (see [I-D.ietf-bfd-multipoint] and [I-D.ietf-bfd-multipoint-active-tail] particularly Section 3.4).

TBD...

6. BFD Packet Encapsulation

6.1 Using GAL/G-ACh Encapsulation Without IP Headers

This section describes use of the Generic Associated Channel Label (GAL/G-ACh).

6.1.1 Ingress Replication

The packet contains the following labels: LSP label (transport) when not using PHP (Penultimate Hop Popping), the optional entropy label, the BUM label and the SH label [RFC7432] (where applicable). The G-ACh type is set to TBD1. The G-Ach payload of the packet MUST contain the L2 header (in overlay space) followed by the IP header encapsulating the BFD packet. The MAC address of the inner packet is used to validate the <EVI, MAC> in the receiving node. The discriminator values of BFD are obtained through negotiation through the out-of-band EVPN LSP ping.

6.1.1.1 Alternative Encapsulation Format

A new TLV can be defined as proposed in Sec 3 of [RFC6428] to include the EVPN FEC information as a TLV following the BFD Control packet.

The format of the TLV can be reused from the EVPN Inclusive Multicast sub-TLV proposed by Fig 2 of [I-D.jain-bess-evpn-lsp-ping].

A new type (TBD3) to indicate the EVPN Inclusive Multicast SubTLV is requested from the "CC/ CV MEP-ID TLV" registry [RFC6428].

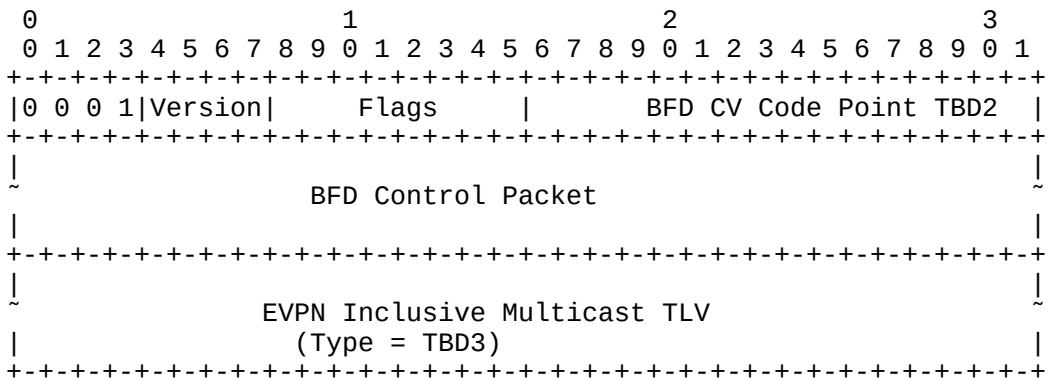


Figure 1: BFD-EVPN CV Message for EVPN Multicast
(Ingress Replication)

6.1.2 LSM (Label Switched Multicast)

TBD...

6.1.3 Unicast

The packet contains the following labels: LSP label (transport) when not using PHP, the optional entropy label and the EVPN Unicast label. The G-ACh type is set to TBD1. The G-Ach payload of the packet MUST contain the L2 header (in overlay space) followed by the IP header encapsulating the BFD packet. The MAC address of the inner packet is used to validate the <EVI, MAC> in the receiving node. The discriminator values for BFD are obtained through negotiation using the out-of-band EVPN ping.

6.1.3.1 Alternative Encapsulation Format

A new TLV can be defined as proposed in Sec 3 of [RFC6428] to include the EVPN FEC information as a TLV following the BFD Control packet. The format of the TLV can be reused from the EVPN MAC sub-TLV proposed by Figure 1 of [I-D.jain-bess-evpn-lsp-ping]. A new type (TBD4) to indicate the EVPN MAC SubTLV is requested from the "CC/ CV MEP-ID TLV" registry [RFC6428].

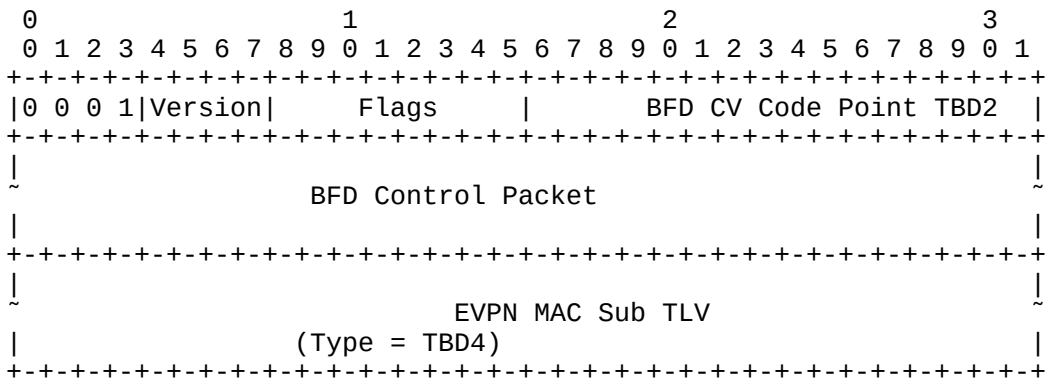


Figure 2: BFD-EVPN CV Message for EVPN Unicast

6.2 Using IP Headers

The encapsulation option using IP headers will not be suited for EVPN, as using different values in the destination IP address for data and OAM (BFD) packets could cause the BFD packets to follow a different path than that of data packets. Hence this option MUST NOT be used for EVPN.

7. Scalability Considerations

The mechanisms proposed by this draft could affect the packet load on the network and its elements especially when supporting configurations involving a large number of EVIs. The option of slowing down or speeding up BFD timer values can be used by an administrator or a network management entity to maintain the overhead incurred due to fault monitoring at an acceptable level.

8. IANA Considerations

IANA is requested to assign two channel types from the "Pseudowire Associated Channel Types" registry in [RFC4385] as follows.

Value	Description	Reference
TBD1	EFD-EVPN CC	[this document]
TBD2	BFD-EVPN CV	[this document]

Ed Note: Do we need a CC code point? TBD

IANA is requested to assign the following code-points from the "CC/CV MEP-ID TLV" registry [RFC6428].

Value	Name	Reference
TBD3	EVPN inclusive multicast	[this document]
TBD4	EVPN unicast	[this document]

9. Security Considerations

Security considerations discussed in [RFC5880], [RFC5883], and [RFC8029] apply.

MPLS security considerations [RFC5920] apply to BFD Control packets encapsulated in a MPLS label stack. When BPD Control packets are routed, the authentication considerations discussed in [RFC5883] should be followed.

Normative References

[I-D.ietf-bess-evpn-inter-subnet-forwarding] Sajassi, A., Salam, S., Thoria, S., Rekhter, Y., Drake, J., Yong, L., and L. Dunbar, "Integrated Routing and Bridging in EVPN", draft-ietf-bess-evpn-inter-subnet-forwarding-03 (work in progress), October 2015.

[I-D.ietf-bfd-multipoint] Katz, D., Ward, D., and J. Networks, "BFD for Multipoint Networks", draft-ietf-bfd-multipoint-16 (work in progress), April 2016.

[I-D.ietf-bfd-multipoint-active-tail] Katz, D., Ward, D., and J. Networks, "BFD Multipoint Active Tails.", draft-ietf-bfd-multipoint-active-tail-07 (work in progress), May 2016.

[I-D.jain-bess-evpn-lsp-ping] Jain, P., Boutros, S., and S. Salam, "LSP-Ping Mechanisms for EVPN and PBB-EVPN", draft-jain-bess-evpn-lsp-ping-06 (work in progress), May 2016.

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

[RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <http://www.rfc-editor.org/info/rfc4385>.

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

[RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <https://www.rfc-editor.org/info/rfc5883>.

[RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <http://www.rfc-editor.org/info/rfc5884>.

[RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", RFC 6425, DOI 10.17487/RFC6425, November 2011, <https://www.rfc-editor.org/info/rfc6425>.

[RFC6428] Allan, D., Ed., Swallow, G., Ed., and J. Drake, Ed., "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, DOI 10.17487/RFC6428, November 2011, <http://www.rfc-editor.org/info/rfc6428>.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <http://www.rfc-editor.org/info/rfc6790>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <http://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <http://www.rfc-editor.org/info/rfc7623>.

[RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <http://www.rfc-editor.org/info/rfc7726>.

[RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <https://www.rfc-editor.org/info/rfc8029>.

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

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

Informative References

[I-D.ooamdt-rtgwg-ooam-requirement] Kumar, N., Pignataro, C., Kumar, D., Mirsky, G., Chen, M., Nordmark, E., Networks, J., and D. Mozes, "Overlay OAM Requirements", draft-ooamdt-rtgwg-

ooam-requirement-02 (work in progress), March 2016.

[I-D.eastlake-bess-evpn-oam-req-frmwk] Salam, S., Sajassi, A., Aldrin, S., and J. Drake, "EVPN Operations, Administration and Maintenance Requirements and Framework", draft-eastlake-bess-evpn-oam-req-frmwk-00 (work in progress), May 2018.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <https://www.rfc-editor.org/info/rfc5920>.

Authors' Addresses

Vengada Prasad Govindan
Cisco Systems

Email: venggovi@cisco.com

Mudigonda Mallik
Cisco Systems

Email: mmudigon@cisco.com

Ali Sajassi
Cisco Systems
170 West Tasman Drive
San Jose, CA 95134, USA

Email: sajassi@cisco.com

Gregory Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Donald Eastlake, 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Copyright, Disclaimer, and Additional IPR Provisions

INTERNET-DRAFT
Intended status: Proposed Standard

V. Govindan
M. Mudigonda
A. Sajassi
Cisco Systems
G. Mirsky
ZTE
D. Eastlake
Futurewei Technologies

Expires: July 1, 2020
January 2, 2020

Fault Management for EVPN networks
draft-gmsm-bess-evpn-bfd-04

Abstract

This document specifies proactive, in-band network OAM mechanisms to detect loss of continuity and miss-connection faults that affect unicast and multi-destination paths (used by Broadcast, Unknown Unicast and Multicast traffic) in an Ethernet VPN (EVPN) network. The mechanisms specified in the draft are based on the widely adopted Bidirectional Forwarding Detection (BFD) protocol.

Status of this Memo

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

Distribution of this document is unlimited. Comments should be sent to the authors or the BESS working group mailing list: bess@ietf.org.

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/1id-abstracts.html. The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Table of Contents

1. Introduction

[ietf-bess-evpn-oam-req-frmwk] outlines the OAM requirements of Ethernet VPN networks (EVPN [RFC7432]). This document specifies mechanisms for proactive fault detection at the network (overlay) layer of EVPN. The mechanisms proposed in the draft use the widely adopted Bidirectional Forwarding Detection (BFD [RFC5880]) protocol.

EVPN fault detection mechanisms need to consider unicast traffic separately from Broadcast, Unknown Unicast, and Multicast (BUM) traffic since they map to different Forwarding Equivalency Classes (FECs) in EVPN. Hence this document proposes different fault detection mechanisms to suit each type, for unicast traffic using BFD [RFC5880] and for BUM traffic using BFD or [RFC8563] depending on whether an MP2P or P2MP tunnel is being used.

Packet loss and packet delay measurement are out of scope for this document.

1.1 Terminology

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

The following acronyms are used in this document.

BFD – Bidirectional Forwarding Detection [RFC5880]

BUM – Broadcast, Unknown Unicast, and Multicast

CC – Continuity Check

CV – Connectivity Verification

EVI – EVPN Instance

EVPN – Ethernet VPN [RFC7432]

FEC – Forwarding Equivalency Class

GAL – Generic Associated Channel Label [RFC5586]

LSM – Label Switched Multicast (P2MP)

LSP – Label Switched Path

MP2P - Multi-Point to Point

OAM - Operations Administration, and Maintenance

P2MP - Point to Multi-Point (LSM)

PE - Provider Edge

VXLAN - Virtual eXtesible Local Area Network (VXLAN) [RFC7348]

2. Scope of this Document

This document specifies BFD based mechanisms for proactive fault detection for EVPN both as specified in [RFC7432] and also for EVPN using VXLAN encapsulation [ietf-vxlan-bfd]. It covers the following:

- Unicast traffic.
- BUM traffic using Multi-point-to-Point (MP2P) tunnels (ingress replication).
- BUM traffic using Point-to-Multipoint (P2MP) tunnels (Label Switched Multicast (LSM)).
- MPLS and VXLAN encapsulation.

This document does not discuss BFD mechanisms for:

- EVPN variants like PBB-EVPN [RFC7623]. It is intended to address this in future versions.
- Integrated Routing and Bridging (IRB) solution based on EVPN [ietf-bess-evpn-inter-subnet-forwarding]. It is intended to address this in future versions.
- EVPN using other encapsulations such as NVGRE or MPLS over GRE [RFC8365].
- BUM traffic using MP2MP tunnels.

This specification specifies procedures for BFD asynchronous mode. BFD demand mode is outside the scope of this specification except as it is used in [RFC8563]. The use of the Echo function is outside the scope of this specification.

3. Motivation for Running BFD at the EVPN Network Layer

The choice of running BFD at the network layer of the OAM model for EVPN [ietf-bess-evpn-oam-req-frmwk] was made after considering the following:

- In addition to detecting link failures in the EVPN network, BFD sessions at the network layer can be used to monitor the successful setup of MP2P and P2MP EVPN tunnels transporting Unicast and BUM traffic such as label programming. The scope of reachability detection covers the ingress and the egress EVPN PE nodes and the network connecting them.

- Monitoring a representative set of path(s) or a particular path among the multiple paths available between two EVPN PE nodes could be done by exercising entropy mechanisms such as entropy labels, when they are used, or VXLAN source ports. However, paths that cannot be realized by entropy variations cannot be monitored. Fault monitoring requirements outlined by [ietf-bess-evpn-oam-req-frmwk] are addressed by the mechanisms proposed by this draft.

BFD testing between EVPN PE nodes does not guarantee that the EVPN service is functioning. (This can be monitored at the service level, that is CE to CE.) For example, an egress EVPN-PE could understand EVPN labeling received but could switch data to an incorrect interface. However, BFD testing in the EVPN Network Layer does provide additional confidence that data transported using those tunnels will reach the expected egress node. When BFD testing in the EVPN overlay fails, that can be used as an indication of a Loss-of-Connectivity defect in the EVPN underlay that would cause EVPN service failure.

4. Fault Detection for Unicast Traffic

The mechanisms specified in BFD for MPLS LSPs [RFC5884] [RFC7726] are applied to test the handling of unicast EVPN traffic. The discriminators required for de-multiplexing the BFD sessions are advertised through BGP as specified in Section 7. This is needed for MPLS since the label stack does not contain enough information to disambiguate the sender of the packet.

The usage of MPLS entropy labels or various VXLAN source ports takes care of the requirement to monitor various paths of the multi-path server layer network [RFC6790]. Each unique realizable path between the participating PE routers MAY be monitored separately when such entropy is used. At least one path of multi-path connectivity between two PE routers MUST be tracked with BFD, but in that case the granularity of fault-detection will be coarser. To support unicast OAM, each PE node MUST allocate a BFD discriminator to be used for BFD messages to that PE and MUST advertise this discriminator with BGP as specified in Section 7. Once the BFD session for the EVPN label is UP, the ends of the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first bring down the session as specified in [RFC5884].

5. Fault Detection for BUM Traffic

Section 5.1 below discusses fault detection for MP2P tunnels using ingress replication and Section 5.2 discusses fault detection for P2MP tunnels.

5.1 Ingress Replication

Ingress replication uses separate MP2P tunnels for transporting BUM traffic from the ingress PE (head) to a set of one or more egress PEs (tails). The fault detection mechanism specified by this document takes advantage of the fact that the head makes a unique copy for each tail.

Another key aspect to be considered in EVPN is the advertisement of the inclusive multicast route. The BUM traffic flows from a head node to a particular tail only after the head receives the inclusive multicast route. This contains the BUM EVPN label (downstream allocated) corresponding to the MP2P tunnel for MPLS encapsulation and contains the IP address of the PE originating the inclusive multicast route for use in VXLAN encapsulation.

There MAY exist multiple BFD sessions between a head PE and an individual tail due to (1) the usage of MPLS entropy labels [RFC6790] or VXLAN source ports for an inclusive multicast FEC and (2) due to multiple MP2P tunnels indicated by different tail labels or IP addresses for MPLS or VXLAN. The BFD discriminator to be used is distributed by BGP as specified in Section 7. Once the BFD session for the EVPN label is UP, the BFD systems terminating the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first bring down the session as specified in [RFC5884].

5.2 P2MP Tunnels (Label Switched Multicast)

Fault detection for BUM traffic distributed using a P2MP tunnel uses active tail multipoint BFD [RFC8563] in one of the three scenarios providing head notification (see Section 5.2 of [RFC8563]).

For MPLS encapsulation of the head to tails BFD, Label Switched Multicast is used. For VXLAN encapsulation, BFD is delivered to the tails through underlay multicast using an outer multicast IP address.

6. BFD Packet Encapsulation

The sections below describe the MPLS and VXLAN encapsulations of BFD for EVPN OAM use.

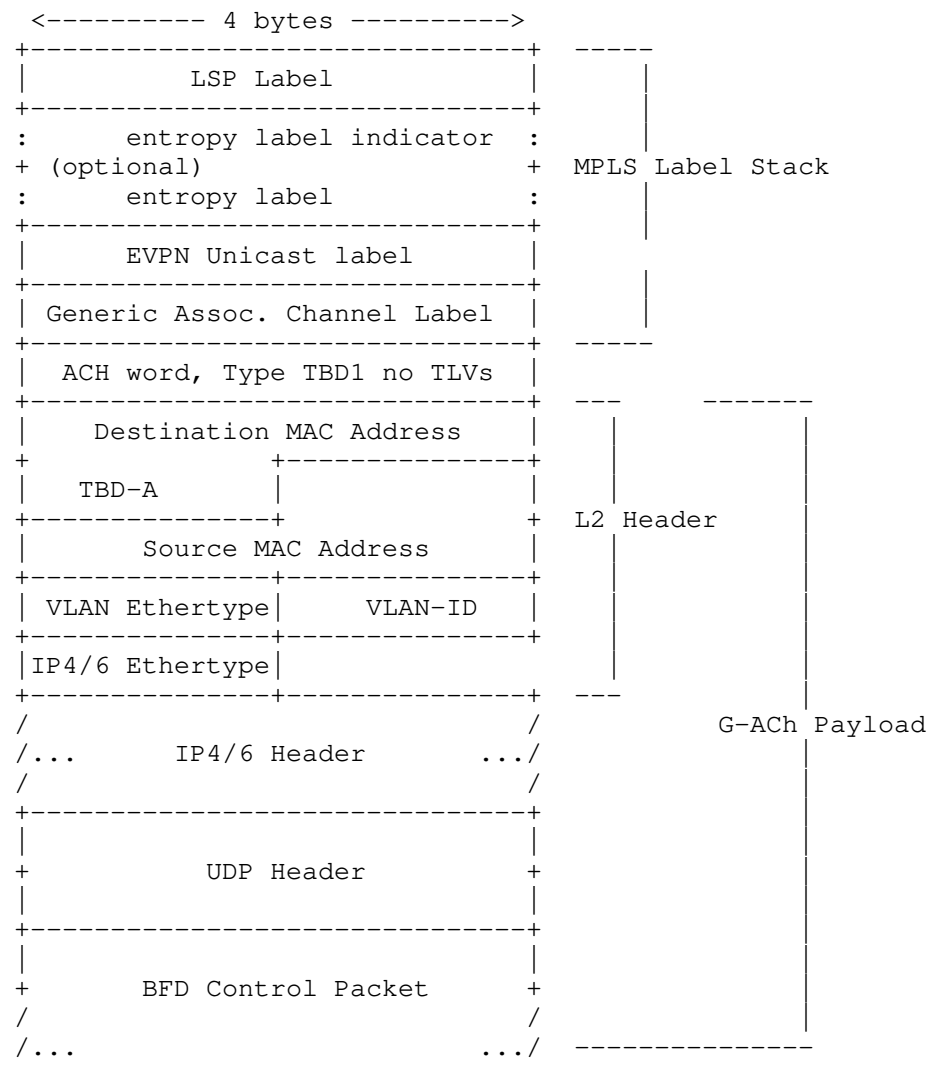
6.1 MPLS Encapsulation

This section describes use of the Generic Associated Channel Label (GAL) for BFD encapsulation in MPLS based EVPN OAM.

6.1.1 Unicast

The packet initially contains the following labels: LSP label (transport), the optional entropy label, and the EVPN Unicast label. The G-ACh type is set to TBD1. The G-ACh payload of the packet MUST contain the destination L2 header (in overlay space) followed by the IP header that encapsulates the BFD packet. The MAC address of the inner packet is used to validate the <EVI, MAC> in the receiving node.

- The destination MAC MUST be the dedicated MAC TBD-A (see Section 9) or the MAC address of the destination PE.
- The destination IP address MUST be in the 127.0.0.0/8 range for IPv4 or in the 0:0:0:0:0:FFFF:7F00:0/104 range for IPv6.
- The destination IP port MUST be 3784 [RFC5881].
- The source IP port MUST be in the range 49152 through 65535.
- The discriminator values for BFD are obtained through BGP as specified in Section 7 or are exchanged out-of-band or through some other means outside the scope of this document.



6.1.2 Ingress Replication

The packet initially contains the following labels: LSP label (transport), the optional entropy label, the BUM label, and the split horizon label [RFC7432] (where applicable). The G-ACh type is set to TBD1. The G-ACh payload of the packet is as described in Section 6.1.1.

6.1.3 LSM (Label Switched Multicast, P2MP)

The encapsulation is the same as in Section 6.1.2 for ingress replication except that the transport label identifies the P2MP tunnel, in effect the set of tail PEs, rather than identifying a single destination PE at the end of an MP2P tunnel.

6.2 VXLAN Encapsulation

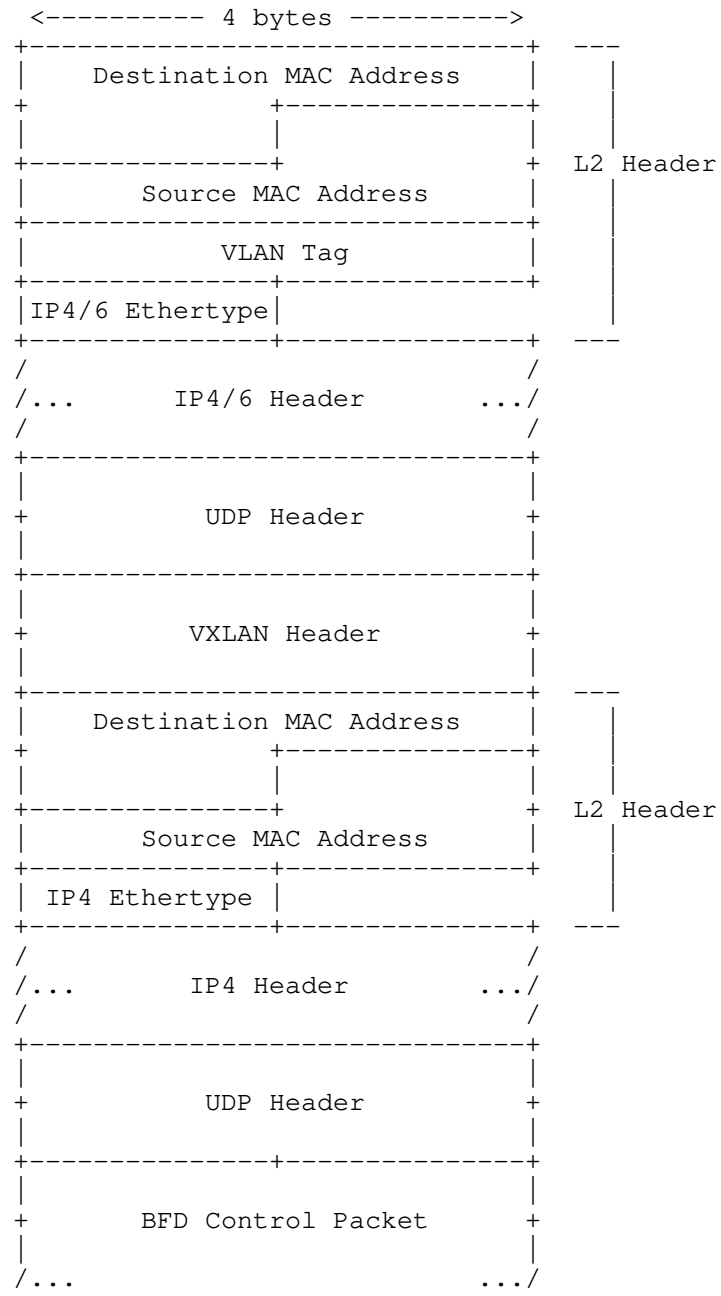
This section describes the use of the VXLAN [RFC7348] for BFD encapsulation in VXLAN based EVPN OAM. This specification conforms to [ietf-bfd-vxlan].

6.2.1 Unicast

The outer and inner IP headers have a unicast source IP address of the BFD message source and a destination IP address of the BFD message destination

The destination UDP port MUST be 3784 [RFC5881]. The source port MUST be in the range 49152 through 65535. If the BFD source has multiple IP addresses, entropy MAY be further obtained by using any of those addresses assuming the source is prepared for responses directed to the IP address used.

The Your BFD discriminator is the value distributed for this unicast OAM purpose by the destination using BGP as specified in Section 7 or is exchanged out-of-band or through some other means outside the scope of this document.


<---------- 4 bytes ---------->
Destination MAC Address
Source MAC Address
VLAN Tag
IP4/6 Ethertype
L2 Header
IP4/6 Header
UDP Header
VXLAN Header
Destination MAC Address
Source MAC Address
IP4 Ethertype
L2 Header
IP4 Header
UDP Header
BFD Control Packet

6.2.2 Ingress Replication

The BFD packet construction is as given in Section 6.2.1 except as follows:

(1) The destination IP address used by the BFD message source is that advertised by the destination PE in its Inclusive Multicast EVPN route for the MP2P tunnel in question; and

(2) The Your BFD discriminator used is the one advertised by the BFD destination using BGP as specified in Section 7 for the MP2P tunnel in question or is exchanged out-of-band or through some other means outside the scope of this document.

6.2.3 LSM (Label Switched Multicast, P2MP)

The VXLAN encapsulation for the head-to-tails BFD packets uses the multicast destination IP corresponding to the VXLAN VNI.

The destination port MUST be 3784. For entropy purposes, the source port can vary but MUST be in the range 49152 through 65535 [RFC5881]. If the head PE has multiple IP addresses, entropy MAY be further obtained by using any of those addresses.

The Your BFD discriminator is the value distributed for this unicast OAM purpose by the BFD message using BGP as specified in Section 7 or is exchanged out-of-band or through some other means outside the scope of this document.

7. BGP Distribution of BFD Discriminators

BGP is used to distribute BFD discriminators for use in EVPN OAM as follows using the BGP-BFD Attribute as specified in [ietf-bess-mvpn-fast-failover]. This attribute is included with appropriate EVPN routes as follows:

Unicast: MAC/IP Advertisement Route [RFC7432].

MP2P Tunnel: Inclusive Multicast Ethernet Tag Route [RFC7432].

P2MP: TBD

[Need more text on BFD sessions reacting to the new advertisement and withdrawal of the BGP-BFD Attribute.]

8. Scalability Considerations

The mechanisms proposed by this draft could affect the packet load on the network and its elements especially when supporting configurations involving a large number of EVIs. The option of slowing down or speeding up BFD timer values can be used by an administrator or a network management entity to maintain the overhead incurred due to fault monitoring at an acceptable level.

9. IANA Considerations

The following IANA Actions are requested.

9.1 Pseudowire Associated Channel Type

IANA is requested to assign a channel type from the "Pseudowire Associated Channel Types" registry in [RFC4385] as follows.

Value	Description	Reference
TBD1	BFD-EVPN OAM	[this document]

9.2 MAC Address

IANA is requested to assign a multicast MAC address under the IANA OUI [0x01005E900004 suggested] as follows:

Address	Usage	Reference
TBD-A	EVPN OAM	[this document]

10. Security Considerations

Security considerations discussed in [RFC5880], [RFC5883], and [RFC8029] apply.

MPLS security considerations [RFC5920] apply to BFD Control packets encapsulated in a MPLS label stack. When BPD Control packets are routed, the authentication considerations discussed in [RFC5883] should be followed.

VXLAN BFD security considerations in [ietf-vxlan-bfd] apply to BFD packets encapsulate in VXLAN.

Acknowledgement

The authors wish to thank the following for their comments and suggestions:

Mach Chen

Normative References

[ietf-bess-evpn-inter-subnet-forwarding] Sajassi, A., Salam, S., Thoria, S., Rekhter, Y., Drake, J., Yong, L., and L. Dunbar, "Integrated Routing and Bridging in EVPN", draft-ietf-bess-evpn-inter-subnet-forwarding-08, work in progress, March 2019.

[ietf-bess-mvpn-fast-failover] Morin, T., Kebler, R., Mirsky, G., "Multicast VPN fast upstream failover", draft-ietf-bess-mvpn-fast-failover-05 (work in progress), February 2019.

[ietf-bfd-vxlan] Pallagatti, S., Paragiri, S., Govindan, V., Mudigonda, M., G. Mirsky, "BFD for VXLAN", draft-ietf-bfd-vxlan-07 (work in progress), May 2019.

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

[RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <http://www.rfc-editor.org/info/rfc4385>.

[RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <https://www.rfc-editor.org/info/rfc5586>.

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

[RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <https://www.rfc-editor.org/info/rfc5881>.

[RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <https://www.rfc-editor.org/info/rfc5883>.

[RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <https://www.rfc-editor.org/info/rfc5884>.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <http://www.rfc-editor.org/info/rfc6790>.

[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, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <http://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <http://www.rfc-editor.org/info/rfc7623>.

[RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <https://www.rfc-editor.org/info/rfc7726>.

[RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <https://www.rfc-editor.org/info/rfc8029>.

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

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

[RFC8563] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) Multipoint Active Tails", RFC 8563, DOI 10.17487/RFC8563, April 2019, <https://www.rfc-editor.org/info/rfc8563>.

Informative References

[ietf-bess-evpn-oam-req-frmwk] Salam, S., Sajassi, A., Aldrin, S., J. Drake, and D. Eastlake, "EVPN Operations, Administration and Maintenance Requirements and Framework", draft-ietf-bess-evpn-oam-req-frmwk-00, work in progress, February 2019.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <https://www.rfc-editor.org/info/rfc5920>.

Authors' Addresses

Vengada Prasad Govindan
Cisco Systems

Email: venggovi@cisco.com

Mudigonda Mallik
Cisco Systems

Email: mmudigon@cisco.com

Ali Sajassi
Cisco Systems
170 West Tasman Drive
San Jose, CA 95134, USA

Email: sajassi@cisco.com

Gregory Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Donald Eastlake, 3rd
Futurewei Technologies
2386 Panoramic Circle
Apopka, FL 32703 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

BESS Working Group
Internet Draft
Intended Status: Proposed Standard
Expires: April 25, 2019

P. Brissette Ed.
Cisco System
H. Shah Ed.
Ciena Corporation
I. Chen Ed.
Jabil
I. Hussain Ed.
Infinera Corporation
K. Tiruveedhula Ed.
Juniper Networks
J. Rabadan Ed.
Nokia

October 22, 2018

Yang Data Model for EVPN
draft-ietf-bess-evpn-yang-06

Abstract

This document describes a YANG data model for Ethernet VPN services. The model is agnostic of the underlay. It apply to MPLS as well as to VxLAN encapsulation. The model is also agnostic of the services including E-LAN, E-LINE and E-TREE services. This document mainly focuses on EVPN and Ethernet-Segment instance framework.

Convention

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. Introduction

The Network Configuration Protocol (NETCONF) [RFC6241] is a network management protocol that defines mechanisms to manage network devices. YANG [RFC6020] is a modular language that represents data structures in an XML or JSON tree format, and is used as a data modeling language for the NETCONF.

This document introduces a YANG data model for Ethernet VPN services (EVPN) [RFC7432], Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN) [RFC7623] as well as other WG draft such as EVPN-VPWS, etc. The EVPN services runs over MPLS and VxLAN underlay.

The Yang data model in this document defines Ethernet VPN based services. The model leverages the definitions used in other IETF Yang draft such as L2VPN Yang.

The goal is to propose a data object model consisting of building blocks that can be assembled in different order to realize different EVPN-based services. The definition work is undertaken initially by a smaller working group with members representing various vendors and service providers. The EVPN basic framework consist of two modules: EVPN and Ethernet-Segment. These models are completely orthogonal. They usually work in pair but user can definitely use one or the other for its own need.

The data model is defined for following constructs that are used for managing the services:

- Configuration
- Operational State
- Notifications

The document is organized to first define the data model for the configuration, operational state and notifications of EVPN and Ethernet-Segment.

The EVPN data object model defined in this document uses the instance centric approach whereby EVPN service attributes are specified for a given EVPN instance.

The Ethernet-Segment data object model defined in this document refer to a specific interface. That interface can be a physical interface, a bundle interface or virtual interface. The latter includes attachment-circuit and pseudowire. The purpose of creating a separate module is due to the fact that it can be used without having the need to have EVPN configured as layer 2/3 service. For example, an access node can be dual-homed to two service nodes servicing a VPLS or an IPVPN core. The access connectivity can be represented by an Ethernet-Segment where EVPN BGP DF election is performed over both service nodes.

2. Specification of Requirements

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. EVPN YANG Model

3.1. Overview

Two top level module, Ethernet-Segment and EVPN, are defined. The Ethernet-Segment contains a list of interface to which any Ethernet-Segment attributes are configured/applied.

The EVPN module has two main containers: common and instance. The first one has common attributes to all VPNs where as the latter has attributes specific to an EVI (EVPN instance). This document state the scope of the EVPN object models definition. The following documents are within the scope. This is not an exhaustive list but a representation of documents that are covered for this work:

- Reqs for EVPN:[RFC7209]
- EVPN: [RFC7432]
- PBB-EVPN: [RFC7623]
- EVPN-VPWS: [RFC8214]
- EVPN-ETREE: [RFC8317]
- EVPN Overlay [RFC8365]

The integration with L2VPN instance Yang model is being done as part of the L2VPN Yang model.
Following documents will be covered at that time:

- (PBB-)EVPN Seamless Integration with (PBB-)VPLS:
draft-ietf-bess-evpn-vpls-seamless-integ
- EVPN Virtual Ethernet Segment:
draft-sajassi-bess-evpn-virtual-eth-segment
- IP Prefix Advertisement in EVPN:
draft-ietf-bess-evpn-prefix-advertisement
- VXLAN DCI Using EVPN:
draft-boutros-l2vpn-vxlan-evpn
- Interconnect Solution for EVPN Overlay networks:
draft-ietf-bess-dci-evpn-overlay
- Integrated Routing and Bridging in EVPN:
draft-ietf-bess-evpn-inter-subnet-forwarding

3.2 Ethernet-Segment Model

The Ethernet-Segment data model has a list of ES where each refer to an interface. All attributes are optional due to auto-sensing default mode where all values are auto-derive from the network connectivity.

module: ietf-ethernet-segment

```
      +--rw ethernet-segments
         +--rw ethernet-segment* [name]
            +--rw name                              string
            +--ro service-type?                     string
            +--ro status?                           status-type
            +--rw (ac-or-pw)?
            |  +--:(ac)
            |  |  +--rw ac*                              if:interface-ref
            |  +--:(pw)
            |     +--rw pw*                           pw:pseudowire-ref
            +--ro interface-status?                 status-type
            +--rw ethernet-segment-identifier?      uint32
            +--rw (active-mode)
            |  +--:(single-active)
            |  |  +--rw single-active-mode?              empty
            |  +--:(all-active)
            |     +--rw all-active-mode?              empty
            +--rw pbb-parameters {ethernet-segment-pbb-params}?
            |  +--rw backbone-src-mac?   yang:mac-address
            +--rw bgp-parameters
            |  +--rw common
            |     +--rw rd-rt* [route-distinguisher]
            |                  {ethernet-segment-bgp-params}?
            |        +--rw route-distinguisher
            |              rt-types:route-distinguisher
            |        +--rw vpn-target* [route-target]
            |           +--rw route-target
            |                 rt-types:route-target
            |           +--rw route-target-type
            |                 rt-types:route-target-type
            +--rw df-election
            |  +--rw df-election-method?   df-election-method-type
            |  +--rw preference?           uint16
            |  +--rw revertive?            boolean
            |  +--rw election-wait-time?   uint32
            +--rw ead-evi-route?                boolean
            +--ro esi-label?                    string
            +--ro member*
            |  +--ro ip-address?   inet:ip-address
            +--ro df*
               +--ro service-identifier?   uint32
               +--ro vlan?                 uint32
               +--ro ip-address?           inet:ip-address
```

3.3 EVPN Model

The evpn-instances container contains a list of evpn-instance. Each entry of the evpn-instance represents a different Ethernet VPN and it

is represented by a EVI. Again, mainly all attributes are optional for the same reason as for the Ethernet-Segment module.

```
module: ietf-evpn
    +--rw evpn
       +--rw common
       |  +--rw (replication-type)?
       |     +--:(ingress-replication)
       |     |  +--rw ingress-replication?   boolean
       |     +--:(p2mp-replication)
       |        +--rw p2mp-replication?      boolean
       +--rw evpn-instances
          +--rw evpn-instance* [name]
             +--rw name                                  string
             +--rw evi?                                  uint32
             +--rw pbb-parameters {evpn-pbb-params}?
             |  +--rw source-bmac?   yang:hex-string
             +--rw bgp-parameters
             |  +--rw common
             |     +--rw rd-rt* [route-distinguisher]
             |                  {evpn-bgp-params}?
             |        +--rw route-distinguisher
             |        |     rt-types:route-distinguisher
             |        +--rw vpn-target* [route-target]
             |           +--rw route-target
             |                        rt-types:route-target
             |           +--rw route-target-type
                                      rt-types:route-target-type
             +--rw arp-proxy?                            boolean
             +--rw arp-suppression?                      boolean
             +--rw nd-proxy?                             boolean
             +--rw nd-suppression?                       boolean
             +--rw underlay-multicast?                   boolean
             +--rw flood-unknown-unicast-supression?     boolean
             +--rw vpws-vlan-aware?                      boolean
             +--ro routes
             |  +--ro ethernet-auto-discovery-route*
             |  |  +--ro rd-rt* [route-distinguisher]
             |  |  |  +--ro route-distinguisher
             |  |  |        rt-types:route-distinguisher
             |  |  |  +--ro vpn-target* [route-target]
             |  |  |     +--ro route-target    rt-types:route-target
             |  |  +--ro ethernet-segment-identifier?   uint32
             |  |  +--ro ethernet-tag?                  uint32
             |  |  +--ro path*
             |  |     +--ro next-hop?   inet:ip-address
             |  |     +--ro label?      rt-types:mpls-label
             |  |     +--ro detail
```

```
|  |        +--ro attributes
|  |        |  +--ro extended-community*   string
|  |        +--ro bestpath?     empty
|  +--ro mac-ip-advertisement-route*
|  |  +--ro rd-rt* [route-distinguisher]
|  |  |  +--ro route-distinguisher
|  |  |         rt-types:route-distinguisher
|  |  |  +--ro vpn-target* [route-target]
|  |  |     +--ro route-target
|  |  |           rt-types:route-target
|  |  +--ro ethernet-segment-identifier?   uint32
|  |  +--ro ethernet-tag?                  uint32
|  |  +--ro mac-address?                 yang:hex-string
|  |  +--ro mac-address-length?            uint8
|  |  +--ro ip-prefix?                   inet:ip-prefix
|  |  +--ro path*
|  |     +--ro next-hop?   inet:ip-address
|  |     +--ro label?      rt-types:mpls-label
|  |     +--ro label2?     rt-types:mpls-label
|  |        +--ro detail
|  |        +--ro attributes
|  |        |  +--ro extended-community*   string
|  |        +--ro bestpath?     empty
|  +--ro inclusive-multicast-ethernet-tag-route*
|  |  +--ro rd-rt* [route-distinguisher]
|  |  |  +--ro route-distinguisher
|  |  |         rt-types:route-distinguisher
|  |  |  +--ro vpn-target* [route-target]
|  |  |     +--ro route-target
|  |  |           rt-types:route-target
|  |  +--ro ethernet-segment-identifier?   uint32
|  |  +--ro originator-ip-prefix?        inet:ip-prefix
|  |  +--ro path*
|  |     +--ro next-hop?   inet:ip-address
|  |     +--ro label?      rt-types:mpls-label
|  |     +--ro detail
|  |        +--ro attributes
|  |        |  +--ro extended-community*   string
|  |        +--ro bestpath?     empty
|  +--ro ethernet-segment-route*
|  |  +--ro rd-rt* [route-distinguisher]
|  |  |  +--ro route-distinguisher
|  |  |         rt-types:route-distinguisher
|  |  |  +--ro vpn-target* [route-target]
|  |  |     +--ro route-target
|  |  |           rt-types:route-target
|  |  +--ro ethernet-segment-identifier?   uint32
|  |  +--ro originator-ip-prefix?        inet:ip-prefix
```

```
                |  |  +--ro path*
                |  |     +--ro next-hop?   inet:ip-address
                |  |     +--ro detail
                |  |        +--ro attributes
                |  |        |  +--ro extended-community*   string
                |  |        +--ro bestpath?     empty
                |  +--ro ip-prefix-route*
                |     +--ro rd-rt* [route-distinguisher]
                |     |  +--ro route-distinguisher
                |     |        rt-types:route-distinguisher
                |     |  +--ro vpn-target* [route-target]
                |     |     +--ro route-target    rt-types:route-target
                |     +--ro ethernet-segment-identifier?   uint32
                |     +--ro ip-prefix?                   inet:ip-prefix
                |     +--ro path*
                |        +--ro next-hop?   inet:ip-address
                |        +--ro label?      rt-types:mpls-label
                |        +--ro detail
                |           +--ro attributes
                |           |  +--ro extended-community*   string
                |           +--ro bestpath?     empty
                +--ro statistics
                   +--ro tx-count?   uint32
                   +--ro rx-count?   uint32
                   +--ro detail
                      +--ro broadcast-tx-count?         uint32
                      +--ro broadcast-rx-count?         uint32
                      +--ro multicast-tx-count?         uint32
                      +--ro multicast-rx-count?         uint32
                      +--ro unknown-unicast-tx-count?   uint32
                      +--ro unknown-unicast-rx-count?   uint32
augment /pw:pseudowires/pw:pseudowire/pw:pw-type:
  +--:(evpn-pw)
     +--rw evpn-pw
        +--rw remote-id?   uint32
        +--rw local-id?    uint32
augment
/ni:network-instances/ni:network-instance/ni:ni-type/l2vpn:l2vpn:
  +--rw evpn-instance?   evpn-instance-ref
augment
/ni:network-instances/ni:network-instance/ni:ni-type/l2vpn:l2vpn:
  +--rw vpls-contstraints

notifications:
  +---n evpn-state-change-notification
     +--ro evpn-instance?   evpn-instance-ref
     +--ro state?           identityref
```

4. YANG Module

The EVPN configuration container is logically divided into following high level configuration areas:

4.1 Ethernet Segment Yang Module

```
<CODE BEGINS> file "ietf-ethernet-segment@2018-02-20.yang"
module ietf-ethernet-segment {
  namespace "urn:ietf:params:xml:ns:yang:ietf-ethernet-segment";
  prefix "es";

  import ietf-yang-types {
    prefix "yang";
  }

  import ietf-inet-types {
    prefix "inet";
  }

  import ietf-routing-types {
    prefix "rt-types";
  }

  import ietf-interfaces {
    prefix "if";
  }

  import ietf-pseudowires {
    prefix "pw";
  }

  organization  "ietf";
  contact       "ietf";
  description   "ethernet segment";

  revision "2018-02-20" {
    description " - Change the type of attachment circuit to " +
                "   if:interface-ref " +
                "";
    reference   "";
  }

  revision "2017-10-21" {
    description " - Updated ethernet segment's AC/PW members to " +
                "   accommodate more than one AC or more than one " +
                "   PW " +
                " - Added the new preference based DF election " +
```

```
                  "   method " +
                  " - Referenced pseudowires in the new " +
                  "   ietf-pseudowires.yang model " +
                  " - Moved model to NMDA style specified in " +
                  "   draft-dsdt-nmda-guidelines-01.txt " +
                  "";
    reference     "";
  }

  revision "2017-03-08" {
    description " - Updated to use BGP parameters from " +
                  "   ietf-routing-types.yang instead of from " +
                  "   ietf-evpn.yang " +
                  " - Updated ethernet segment's AC/PW members to " +
                  "   accommodate more than one AC or more than one " +
                  "   PW " +
                  " - Added the new preference based DF election " +
                  "   method " +
                  "";
    reference     "";
  }

  revision "2016-07-08" {
    description "   - Added the configuration option to enable or " +
                "     disable per-EVI/EAD route " +
                "   - Added PBB parameter backbone-src-mac " +
                "   - Added operational state branch, initially " +
                "     to match the configuration branch" +
                "";
    reference   "";
  }

  revision "2016-06-23" {
    description "WG document adoption";
    reference   "";
  }

  revision "2015-10-15" {
    description "Initial revision";
    reference   "";
  }

  /* Features */

  feature ethernet-segment-bgp-params {
    description "Ethernet segment's BGP parameters";
  }
```

```
feature ethernet-segment-pbb-params {
  description "Ethernet segment's PBB parameters";
}

/* Typedefs */
typedef status-type {
  type enumeration {
    enum up {
      description "Status is up";
    }
    enum down {
      description "Status is down";
    }
  }
  description "status type";
}

typedef df-election-method-type {
  type enumeration {
    enum default {
      value 0;
      description "The default DF election method";
    }
    enum highest-random-weight {
      value 1;
      description "The highest random weight (HRW) method";
      reference "draft-mohanty-bess-evpn-df-election";
    }
    enum preference {
      value 2;
      description "The preference based method";
      reference "draft-rabadan-bess-evpn-pref-df";
    }
  }
  description "The DF election method type";
}

/* EVPN Ethernet Segment YANG Model */

container ethernet-segments {
  description "ethernet-segment";
  list ethernet-segment {
    key "name";
    leaf name {
      type string;
      description "Name of the ethernet segment";
    }
    leaf service-type {
```

```
          type string;
          config false;
          description "service-type";
        }
        leaf status {
          type status-type;
          config false;
          description "Ethernet segment status";
        }
        choice ac-or-pw {
          description "ac-or-pw";
          case ac {
            leaf-list ac {
              type if:interface-ref;
              description "Name of attachment circuit";
            }
          }
          case pw {
            leaf-list pw {
              type pw:pseudowire-ref;
              description "Reference to a pseudowire";
            }
          }
        }
        leaf interface-status {
          type status-type;
          config false;
          description "interface status";
        }
        leaf ethernet-segment-identifier {
          type uint32;
          description "Ethernet segment identifier (esi)";
        }
        choice active-mode {
          mandatory true;
          description "Choice of active mode";
          case single-active {
            leaf single-active-mode {
              type empty;
              description "single-active-mode";
            }
          }
          case all-active {
            leaf all-active-mode {
              type empty;
              description "all-active-mode";
            }
          }
```

```
        }
        container pbb-parameters {
          if-feature ethernet-segment-pbb-params;
          description "PBB configuration";
          leaf backbone-src-mac {
            type yang:mac-address;
            description "backbone-src-mac, only if this is a PBB";
          }
        }
        container bgp-parameters {
          description "BGP parameters";
          container common {
            description "BGP parameters common to all pseudowires";
            list rd-rt {
              if-feature ethernet-segment-bgp-params;
              key "route-distinguisher";
              leaf route-distinguisher {
                type rt-types:route-distinguisher;
                description "Route distinguisher";
              }
              uses rt-types:vpn-route-targets;
              description "A list of route distinguishers and " +
                          "corresponding VPN route targets";
            }
          }
        }
        container df-election {
          description "df-election";
          leaf df-election-method {
            type df-election-method-type;
            description "The DF election method";
          }
          leaf preference {
            when "../df-election-method = 'preference'" {
              description "The preference value is only applicable " +
                          "to the preference based method";
            }
            type uint16;
            description "The DF preference";
          }
          leaf revertive {
            when "../df-election-method = 'preference'" {
              description "The revertive value is only applicable " +
                          "to the preference method";
            }
            type boolean;
            default true;
            description "The 'preempt' or 'revertive' behavior";
```

```
          }
          leaf election-wait-time {
            type uint32;
            description "election-wait-time";
          }
        }
        leaf ead-evi-route {
          type boolean;
          default false;
          description "Enable (true) or disable (false) ead-evi-route";
        }
        leaf esi-label {
          type string;
          config false;
          description "esi-label";
        }
        list member {
          config false;
          leaf ip-address {
            type inet:ip-address;
            description "ip-address";
          }
          description "member of the ethernet segment";
        }
        list df {
          config false;
          leaf service-identifier {
            type uint32;
            description "service-identifier";
          }
          leaf vlan {
            type uint32;
            description "vlan";
          }
          leaf ip-address {
            type inet:ip-address;
            description "ip-address";
          }
          description "df of an evpn instance's vlan";
        }
        description "An ethernet segment";
      }
    }
  }
  <CODE ENDS>
```

4.2 EVPN Yang Module

```
<CODE BEGINS> file "ietf-evpn@2018-02-20.yang"
module ietf-evpn {
  namespace "urn:ietf:params:xml:ns:yang:ietf-evpn";
  prefix "evpn";

  import ietf-inet-types {
    prefix "inet";
  }

  import ietf-yang-types {
    prefix "yang";
  }

  import ietf-routing-types {
    prefix "rt-types";
  }

  import ietf-network-instance {
    prefix "ni";
  }

  import ietf-l2vpn {
    prefix "l2vpn";
  }

  import ietf-pseudowires {
    prefix "pw";
  }

  organization  "ietf";
  contact       "ietf";
  description   "evpn";

  revision "2018-02-20" {
    description " - Incorporated ietf-network-instance model" +
                "   on which ietf-l2vpn is now based " +
                "";
    reference   "";
  }

  revision "2017-10-21" {
    description " - Modified the operational state augment " +
                " - Renamed evpn-instances-state to evpn-instances" +
                " - Added vpws-vlan-aware to an EVPN instance " +
                " - Added a new augment to L2VPN to add EPVN " +
                " - pseudowire for the case of EVPN VPWS " +
                " - Added state change notification " +
                "";
```

```
      reference   "";
    }

    revision "2017-03-13" {
      description " - Added an augment to base L2VPN model to " +
                  "   reference an EVPN instance " +
                  " - Reused ietf-routing-types.yang " +
                  "   vpn-route-targets grouping instead of " +
                  "   defining it in this module " +
                  "";
      reference   "";
    }

    revision "2016-07-08" {
      description " - Added operational state" +
                  " - Added a configuration knob to enable/disable " +
                  "   underlay-multicast " +
                  " - Added a configuration knob to enable/disable " +
                  "   flooding of unknonw unicast " +
                  " - Added several configuration knobs " +
                  "   to manage ARP and ND" +
                  "";
      reference   "";
    }

    revision "2016-06-23" {
      description "WG document adoption";
      reference   "";
    }

    revision "2015-10-15" {
      description "Initial revision";
      reference   "";
    }

    feature evpn-bgp-params {
      description "EVPN's BGP parameters";
    }

    feature evpn-pbb-params {
      description "EVPN's PBB parameters";
    }

    /* Identities */

    identity evpn-notification-state {
      description "The base identity on which EVPN notification " +
                  "states are based";
```

```
  }

  identity MAC-duplication-detected {
    base "evpn-notification-state";
    description "MAC duplication is detected";
  }

  identity mass-withdraw-received {
    base "evpn-notification-state";
    description "Mass withdraw received";
  }

  identity static-MAC-move-detected {
    base "evpn-notification-state";
    description "Static MAC move is detected";
  }

  /* Typedefs */

  typedef evpn-instance-ref {
    type leafref {
      path "/evpn/evpn-instances/evpn-instance/name";
    }
    description "A leafref type to an EVPN instance";
  }

  /* Groupings */

  grouping route-rd-rt-grp {
    description "A grouping for a route's route distinguishers " +
                "and route targets";
    list rd-rt {
      key "route-distinguisher";
      leaf route-distinguisher {
        type rt-types:route-distinguisher;
        description "Route distinguisher";
      }
      list vpn-target {
        key "route-target";
        leaf route-target {
          type rt-types:route-target;
          description "BGP route target";
        }
        description "A list of route targets";
      }
      description "A list of route distinguishers and " +
                  "corresponding VPN route targets";
    }
```

```
    }

    grouping next-hop-label-grp {
      description "next-hop-label-grp";
      leaf next-hop {
        type inet:ip-address;
        description "next-hop";
      }
      leaf label {
        type rt-types:mpls-label;
        description "label";
      }
    }

    grouping next-hop-label2-grp {
      description "next-hop-label2-grp";
      leaf label2 {
        type rt-types:mpls-label;
        description "label2";
      }
    }

    grouping path-detail-grp {
      description "path-detail-grp";
      container detail {
        config false;
        description "path details";
        container attributes {
          leaf-list extended-community {
            type string;
            description "extended-community";
          }
          description "attributes";
        }
        leaf bestpath {
          type empty;
          description "Indicate this path is the best path";
        }
      }
    }

    /* EVPN YANG Model */

    container evpn {
      description "evpn";
      container common {
        description "common epn attributes";
        choice replication-type {
```

```
          description "A choice of replication type";
          case ingress-replication {
            leaf ingress-replication {
              type boolean;
              description "ingress-replication";
            }
          }
          case p2mp-replication {
            leaf p2mp-replication {
              type boolean;
              description "p2mp-replication";
            }
          }
        }
      }
      container evpn-instances {
        description "evpn-instances";
        list evpn-instance {
          key "name";
          description "An EVPN instance";
          leaf name {
            type string;
            description "Name of EVPN instance";
          }
          leaf evi {
            type uint32;
            description "evi";
          }
          container pbb-parameters {
            if-feature "evpn-pbb-params";
            description "PBB parameters";
            leaf source-bmac {
              type yang:hex-string;
              description "source-bmac";
            }
          }
          container bgp-parameters {
            description "BGP parameters";
            container common {
              description "BGP parameters common to all pseudowires";
              list rd-rt {
                if-feature evpn-bgp-params;
                key "route-distinguisher";
                leaf route-distinguisher {
                  type rt-types:route-distinguisher;
                  description "Route distinguisher";
                }
                uses rt-types:vpn-route-targets;
```

```
            description "A list of route distinguishers and " +
                        "corresponding VPN route targets";
          }
        }
      }
      leaf arp-proxy {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) ARP proxy";
      }
      leaf arp-suppression {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) " +
                    "ARP suppression";
      }
      leaf nd-proxy {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) ND proxy";
      }
      leaf nd-suppression {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) " +
                    "ND suppression";
      }
      leaf underlay-multicast {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) " +
                    "underlay multicast";
      }
      leaf flood-unknown-unicast-supression {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) " +
                    "flood unknown unicast suppression";
      }
      leaf vpws-vlan-aware {
        type boolean;
        default false;
        description "Enable (TRUE) or disable (FALSE) " +
                    "VPWS VLAN aware";
      }
      container routes {
        config false;
        description "routes";
```

```
            list ethernet-auto-discovery-route {
              uses route-rd-rt-grp;
              leaf ethernet-segment-identifier {
                type uint32;
                description "Ethernet segment identifier (esi)";
              }
              leaf ethernet-tag {
                type uint32;
                description "An ethernet tag (etag) indentifying a " +
                            "broadcast domain";
              }
              list path {
                uses next-hop-label-grp;
                uses path-detail-grp;
                description "path";
              }
              description "ethernet-auto-discovery-route";
            }
            list mac-ip-advertisement-route {
              uses route-rd-rt-grp;
              leaf ethernet-segment-identifier {
                type uint32;
                description "Ethernet segment identifier (esi)";
              }
              leaf ethernet-tag {
                type uint32;
                description "An ethernet tag (etag) indentifying a " +
                            "broadcast domain";
              }
              leaf mac-address {
                type yang:hex-string;
                description "Route mac address";
              }
              leaf mac-address-length {
                type uint8 {
                  range "0..48";
                }
                description "mac address length";
              }
              leaf ip-prefix {
                type inet:ip-prefix;
                description "ip-prefix";
              }
              list path {
                uses next-hop-label-grp;
                uses next-hop-label2-grp;
                uses path-detail-grp;
                description "path";
```

```
            }
            description "mac-ip-advertisement-route";
          }
          list inclusive-multicast-ethernet-tag-route {
            uses route-rd-rt-grp;
            leaf ethernet-segment-identifier {
              type uint32;
              description "Ethernet segment identifier (esi)";
            }
            leaf originator-ip-prefix {
              type inet:ip-prefix;
              description "originator-ip-prefix";
            }
            list path {
              uses next-hop-label-grp;
              uses path-detail-grp;
              description "path";
            }
            description "inclusive-multicast-ethernet-tag-route";
          }
          list ethernet-segment-route {
            uses route-rd-rt-grp;
            leaf ethernet-segment-identifier {
              type uint32;
              description "Ethernet segment identifier (esi)";
            }
            leaf originator-ip-prefix {
              type inet:ip-prefix;
              description "originator ip-prefix";
            }
            list path {
              leaf next-hop {
                type inet:ip-address;
                description "next-hop";
              }
              uses path-detail-grp;
              description "path";
            }
            description "ethernet-segment-route";
          }
          list ip-prefix-route {
            uses route-rd-rt-grp;
            leaf ethernet-segment-identifier {
              type uint32;
              description "Ethernet segment identifier (esi)";
            }
            leaf ip-prefix {
              type inet:ip-prefix;
```

```
              description "ip-prefix";
            }
            list path {
              uses next-hop-label-grp;
              uses path-detail-grp;
              description "path";
            }
            description "ip-prefix route";
          }
        }
        container statistics {
          config false;
          description "Statistics";
          leaf tx-count {
            type uint32;
            description "transmission count";
          }
          leaf rx-count {
            type uint32;
            description "receive count";
          }
          container detail {
            description "Detailed statistics";
            leaf broadcast-tx-count {
              type uint32;
              description "broadcast transmission count";
            }
            leaf broadcast-rx-count {
              type uint32;
              description "broadcast receive count";
            }
            leaf multicast-tx-count {
              type uint32;
              description "multicast transmission count";
            }
            leaf multicast-rx-count {
              type uint32;
              description "multicast receive count";
            }
            leaf unknown-unicast-tx-count {
              type uint32;
              description "unknown unicast transmission count";
            }
            leaf unknown-unicast-rx-count {
              type uint32;
              description "unknown-unicast receive count";
            }
          }
```

```
        }
      }
    }
  }

  /* augments */

  augment "/pw:pseudowires/pw:pseudowire/pw:pw-type" {
    description "Augment for an L2VPN instance to add EVPN VPWS " +
                "pseudowire";
    case evpn-pw {
      container evpn-pw {
        description "EVPN pseudowire";
        leaf remote-id {
          type uint32;
          description "Remote pseudowire ID";
        }
        leaf local-id {
          type uint32;
          description "Local pseudowire ID";
        }
      }
    }
  }

  augment "/ni:network-instances/ni:network-instance/ni:ni-type" +
          "/l2vpn:l2vpn" {
    description "Augment for an L2VPN instance and EVPN association";
    leaf evpn-instance {
      type evpn-instance-ref;
      description "Reference to an EVPN instance";
    }
  }

  augment "/ni:network-instances/ni:network-instance/ni:ni-type" +
          "/l2vpn:l2vpn" {
    when "l2vpn:type = 'l2vpn:vpls-instance-type'" {
      description "Constraints only for VPLS pseudowires";
    }
    description "Augment for VPLS instance";
    container vpls-contstraints {
      must "not(boolean(/pw:pseudowires/pw:pseudowire" +
           "                [pw:name = current()/../l2vpn:endpoint" +
           "            /l2vpn:pw/l2vpn:name]" +
           "            /evpn-pw/remote-id)) and " +
           "not(boolean(/pw:pseudowires/pw:pseudowire" +
           "                [pw:name = current()/../l2vpn:endpoint" +
           "            /l2vpn:pw/l2vpn:name]" +
```

```
              "            /evpn-pw/local-id)) and " +
              "not(boolean(/pw:pseudowires/pw:pseudowire" +
              "                [pw:name = current()/../l2vpn:endpoint" +
              "            /l2vpn:primary-pw/l2vpn:name]" +
              "            /evpn-pw/remote-id)) and " +
              "not(boolean(/pw:pseudowires/pw:pseudowire" +
              "                [pw:name = current()/../l2vpn:endpoint" +
              "            /l2vpn:primary-pw/l2vpn:name]" +
              "            /evpn-pw/local-id)) and " +
              "not(boolean(/pw:pseudowires/pw:pseudowire" +
              "                [pw:name = current()/../l2vpn:endpoint" +
              "            /l2vpn:backup-pw/l2vpn:name]" +
              "            /evpn-pw/remote-id)) and " +
              "not(boolean(/pw:pseudowires/pw:pseudowire" +
              "                [pw:name = current()/../l2vpn:endpoint" +
              "            /l2vpn:backup-pw/l2vpn:name]" +
              "            /evpn-pw/local-id))" {
            description "A VPLS pseudowire must not be EVPN PW";
          }
          description "VPLS constraints";
        }
      }

      /* Notifications */

      notification evpn-state-change-notification {
        description "EVPN state change notification";
        leaf evpn-instance {
          type evpn-instance-ref;
          description "Related EVPN instance";
        }
        leaf state {
          type identityref {
            base evpn-notification-state;
          }
          description "State change notification";
        }
      }
    }
    <CODE ENDS>
```

5. Security Considerations

The configuration, state, action and notification data defined in this document are designed to be accessed via the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the secure transport layer and the mandatory-to-implement secure transport is SSH [RFC6242]. The NETCONF access control model [RFC6536] provides means to restrict

access for particular NETCONF users to a pre-configured subset of all available NETCONF protocol operations and content.

The security concerns listed above are, however, no different than faced by other routing protocols. Hence, this draft does not change any underlying security issues inherent in [I-D.ietf-netmod-routing-cfg]

6. IANA Considerations

None.

7. References

7.1. Normative Reference

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

7.2. Informative References

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <https://www.rfc-editor.org/info/rfc6536>.

[RFC7209] Sajassi, A., Aggarwal, R., Uttaro, J., Bitar, N., Henderickx, W., and A. Isaac, "Requirements for Ethernet VPN (EVPN)", RFC 7209, DOI 10.17487/RFC7209, May 2014, <https://www.rfc-editor.org/info/rfc7209>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

Authors' Addresses

Patrice Brissette
Cisco Systems, Inc.
EMail: pbrisset@cisco.com

Himanshu Shah
Ciena Corporation
EMail: hshah@ciena.com

Helen Chen
Jabil
EMail: Ing-Wher_Chen@jabil.com

Iftekar Hussain
Infinera Corporation
EMail: ihussain@infinera.com

Kishore Tiruveedhula
Juniper Networks
EMail: kishoret@juniper.net

Jorge Rabadan
Nokia
EMail: jorge.rabadan@nokia.com

Ali Sajassi
Cisco Systems, Inc.
EMail: sajassi@cisco.com

Zhenbin Li

Huawei Technologies
EMail: lizhenbin@huawei.com

BESS Working Group
Internet Draft
Intended Status: Proposed Standard
Expires: September 12, 2019

P. Brissette Ed.
Cisco System
H. Shah Ed.
Ciena Corporation
I. Chen Ed.
Jabil
I. Hussain Ed.
Infinera Corporation
K. Tiruveedhula Ed.
Juniper Networks
J. Rabadan Ed.
Nokia

March 11, 2019

Yang Data Model for EVPN
draft-ietf-bess-evpn-yang-07

Abstract

This document describes a YANG data model for Ethernet VPN services. The model is agnostic of the underlay. It apply to MPLS as well as to VxLAN encapsulation. The model is also agnostic of the services including E-LAN, E-LINE and E-TREE services. This document mainly focuses on EVPN and Ethernet-Segment instance framework.

Convention

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. Introduction

The Network Configuration Protocol (NETCONF) [RFC6241] is a network management protocol that defines mechanisms to manage network devices. YANG [RFC6020] is a modular language that represents data structures in an XML or JSON tree format, and is used as a data modeling language for the NETCONF.

This document introduces a YANG data model for Ethernet VPN services (EVPN) [RFC7432], Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN) [RFC7623] as well as other WG draft such as EVPN-VPWS, etc. The EVPN services runs over MPLS and VxLAN underlay.

The Yang data model in this document defines Ethernet VPN based services. The model leverages the definitions used in other IETF Yang draft such as L2VPN Yang.

The goal is to propose a data object model consisting of building blocks that can be assembled in different order to realize different EVPN-based services. The definition work is undertaken initially by a smaller working group with members representing various vendors and service providers. The EVPN basic framework consist of two modules: EVPN and Ethernet-Segment. These models are completely orthogonal. They usually work in pair but user can definitely use one or the other for its own need.

The data model is defined for following constructs that are used for managing the services:

- Configuration
- Operational State
- Notifications

The document is organized to first define the data model for the configuration, operational state and notifications of EVPN and Ethernet-Segment.

The EVPN data object model defined in this document uses the instance centric approach whereby EVPN service attributes are specified for a given EVPN instance.

The Ethernet-Segment data object model defined in this document refer to a specific interface. That interface can be a physical interface, a bundle interface or virtual interface. The latter includes attachment-circuit and pseudowire. The purpose of creating a separate module is due to the fact that it can be used without having the need to have EVPN configured as layer 2/3 service. For example, an access node can be dual-homed to two service nodes servicing a VPLS or an IPVPN core. The access connectivity can be represented by an Ethernet-Segment where EVPN BGP DF election is performed over both service nodes.

2. Specification of Requirements

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. EVPN YANG Model

3.1. Overview

Two top level module, Ethernet-Segment and EVPN, are defined. The Ethernet-Segment contains a list of interface to which any Ethernet-Segment attributes are configured/applied.

The EVPN module has two main containers: common and instance. The first one has common attributes to all VPNs where as the latter has attributes specific to an EVI (EVPN instance). This document state the scope of the EVPN object models definition. The following documents are within the scope. This is not an exhaustive list but a representation of documents that are covered for this work:

- Reqs for EVPN:[RFC7209]
- EVPN: [RFC7432]
- PBB-EVPN: [RFC7623]
- EVPN-VPWS: [RFC8214]
- EVPN-ETREE: [RFC8317]
- EVPN Overlay [RFC8365]

The integration with L2VPN instance Yang model is being done as part of the L2VPN Yang model.
Following documents will be covered at that time:

- (PBB-)EVPN Seamless Integration with (PBB-)VPLS:
draft-ietf-bess-evpn-vpls-seamless-integ
- EVPN Virtual Ethernet Segment:
draft-sajassi-bess-evpn-virtual-eth-segment
- IP Prefix Advertisement in EVPN:
draft-ietf-bess-evpn-prefix-advertisement
- VXLAN DCI Using EVPN:
draft-boutros-l2vpn-vxlan-evpn
- Interconnect Solution for EVPN Overlay networks:
draft-ietf-bess-dci-evpn-overlay
- Integrated Routing and Bridging in EVPN:
draft-ietf-bess-evpn-inter-subnet-forwarding

3.2 Ethernet-Segment Model

The Ethernet-Segment data model has a list of ES where each refer to an interface. All attributes are optional due to auto-sensing default mode where all values are auto-derive from the network connectivity.

module: ietf-ethernet-segment

```
       +--rw ethernet-segments
          +--rw ethernet-segment* [name]
             +--rw name                                string
             +--ro service-type?                       string
             +--ro status?                             status-type
             +--rw (ac-or-pw)?
             |  +--:(ac)
             |  |  +--rw ac*                                if:interface-ref
             |  +--:(pw)
             |     +--rw pw*                               pw:pseudowire-ref
             +--ro interface-status?                   status-type
             +--rw ethernet-segment-identifier?   ethernet-segment-identifier-ty
pe
             +--rw (active-mode)
             |  +--:(single-active)
             |  |  +--rw single-active-mode?                empty
             |  +--:(all-active)
             |     +--rw all-active-mode?                   empty
             +--rw pbb-parameters {ethernet-segment-pbb-params}?
             |  +--rw backbone-src-mac?   yang:mac-address
             +--rw bgp-parameters
             |  +--rw common
             |     +--rw rd-rt* [route-distinguisher]
             |                  {ethernet-segment-bgp-params}?
             |        +--rw route-distinguisher
             |              rt-types:route-distinguisher
             |        +--rw vpn-targets
             |              rt-types:vpn-route-targets
             +--rw df-election
             |  +--rw df-election-method?   df-election-method-type
             |  +--rw preference?           uint16
             |  +--rw revertive?            boolean
             |  +--rw election-wait-time?   uint32
             +--rw ead-evi-route?                 boolean
             +--ro esi-label?                     string
             +--ro member*
             |  +--ro ip-address?   inet:ip-address
             +--ro df*
                +--ro service-identifier?   uint32
                +--ro vlan?                 uint32
                +--ro ip-address?           inet:ip-address
```

3.3 EVPN Model

The evpn-instances container contains a list of evpn-instance. Each entry of the evpn-instance represents a different Ethernet VPN and it is represented by a EVI. Again, mainly all attributes are optional for the same reason as for the Ethernet-Segment module.

```
   module: ietf-evpn
       +--rw evpn
          +--rw common
          |  +--rw (replication-type)?
          |     +--:(ingress-replication)
          |     |  +--rw ingress-replication?   boolean
          |     +--:(p2mp-replication)
          |        +--rw p2mp-replication?      boolean
          +--rw evpn-instances
             +--rw evpn-instance* [name]
                +--rw name                                    string
                +--rw evi?                                    uint32
                +--rw pbb-parameters {evpn-pbb-params}?
                |  +--rw source-bmac?   yang:mac-address
                +--rw bgp-parameters
                |  +--rw common
                |     +--rw rd-rt* [route-distinguisher]
                |                  {evpn-bgp-params}?
                |        +--rw route-distinguisher
                |        |     rt-types:route-distinguisher
                |        +--rw vpn-targets
                |              rt-types:vpn-route-targets
                +--rw arp-proxy?                              boolean
                +--rw arp-suppression?                        boolean
                +--rw nd-proxy?                               boolean
                +--rw nd-suppression?                         boolean
                +--rw underlay-multicast?                     boolean
                +--rw flood-unknown-unicast-supression?       boolean
                +--rw vpws-vlan-aware?                        boolean
                +--ro routes
                |  +--ro ethernet-auto-discovery-route*
                |  |  +--ro rd-rt* [route-distinguisher]
                |  |  |  +--ro route-distinguisher
                |  |  |        rt-types:route-distinguisher
                |  |  |  +--ro vpn-targets
                |  |  |        rt-types:vpn-route-targets
                |  |  +--ro ethernet-segment-identifier?   es:ethernet-segment-i
dentifier-type
                |  |  +--ro ethernet-tag?                  uint32
                |  |  +--ro path*
                |  |     +--ro next-hop?   inet:ip-address
                |  |     +--ro label?      rt-types:mpls-label
                |  |     +--ro detail
                |  |        +--ro attributes
                |  |        |  +--ro extended-community*   string
                |  |        +--ro bestpath?     empty
                |  +--ro mac-ip-advertisement-route*
                |  |  +--ro rd-rt* [route-distinguisher]
                |  |  |  +--ro route-distinguisher
```

```
                |  |  |        rt-types:route-distinguisher
                |  |  |  +--ro vpn-targets
                |  |  |        rt-types:vpn-route-targets
                |  |  +--ro ethernet-segment-identifier?   es:ethernet-segment-i
dentifier-type
                |  |  +--ro ethernet-tag?                  uint32
                |  |  +--ro mac-address?                   yang:mac-address
                |  |  +--ro mac-address-length?            uint8
                |  |  +--ro ip-prefix?                     inet:ip-prefix
                |  |  +--ro path*
                |  |     +--ro next-hop?   inet:ip-address
                |  |     +--ro label?      rt-types:mpls-label
                |  |     +--ro label2?     rt-types:mpls-label
                |  |        +--ro detail
                |  |        +--ro attributes
                |  |        |  +--ro extended-community*   string
                |  |        +--ro bestpath?     empty
                |  +--ro inclusive-multicast-ethernet-tag-route*
                |  |  +--ro rd-rt* [route-distinguisher]
                |  |  |  +--ro route-distinguisher
                |  |  |        rt-types:route-distinguisher
                |  |  |  +--ro vpn-targets
                |  |  |        rt-types:vpn-route-targets
                |  |  +--ro originator-ip-prefix?          inet:ip-prefix
                |  |  +--ro path*
                |  |     +--ro next-hop?   inet:ip-address
                |  |     +--ro label?      rt-types:mpls-label
                |  |     +--ro detail
                |  |        +--ro attributes
                |  |        |  +--ro extended-community*   string
                |  |        +--ro bestpath?     empty
                |  +--ro ethernet-segment-route*
                |  |  +--ro rd-rt* [route-distinguisher]
                |  |  |  +--ro route-distinguisher
                |  |  |        rt-types:route-distinguisher
                |  |  |  +--ro vpn-targets
                |  |  |        rt-types:vpn-route-targets
                |  |  +--ro ethernet-segment-identifier?   es:ethernet-segment-i
dentifier-type
                |  |  +--ro originator-ip-prefix?          inet:ip-prefix
                |  |  +--ro path*
                |  |     +--ro next-hop?   inet:ip-address
                |  |     +--ro detail
                |  |        +--ro attributes
                |  |        |  +--ro extended-community*   string
                |  |        +--ro bestpath?     empty
                |  +--ro ip-prefix-route*
                |     +--ro rd-rt* [route-distinguisher]
                |     |  +--ro route-distinguisher
                |     |        rt-types:route-distinguisher
```

```
                |      |  +--ro vpn-targets
                |      |        rt-types:vpn-route-targets
                |      +--ro ethernet-segment-identifier?
                |      |     es:ethernet-segment-identifier-type
                |      +--ro ip-prefix?                    inet:ip-prefix
                |      +--ro path*
                |         +--ro next-hop?   inet:ip-address
                |         +--ro label?      rt-types:mpls-label
                |         +--ro detail
                |            +--ro attributes
                |            |  +--ro extended-community*   string
                |            +--ro bestpath?     empty
                |      +--ro statistics
                |         +--ro tx-count?   yang:zero-based-counter32
                |         +--ro rx-count?   yang:zero-based-counter32
                |         +--ro detail
                |            +--ro broadcast-tx-count?
                |                  yang:zero-based-counter32
                |            +--ro broadcast-rx-count?
                |                  yang:zero-based-counter32
                |            +--ro multicast-tx-count?
                |                  yang:zero-based-counter32
                |            +--ro multicast-rx-count?
                |                  yang:zero-based-counter32
                |            +--ro unknown-unicast-tx-count?
                |                  yang:zero-based-counter32
                |            +--ro unknown-unicast-rx-count?
                |                  yang:zero-based-counter32
     augment /pw:pseudowires/pw:pseudowire/pw:pw-type:
       +--:(evpn-pw)
          +--rw evpn-pw
             +--rw remote-id?   uint32
             +--rw local-id?    uint32
     augment
     /ni:network-instances/ni:network-instance/ni:ni-type/l2vpn:l2vpn:
       +--rw evpn-instance?   evpn-instance-ref
     augment
     /ni:network-instances/ni:network-instance/ni:ni-type/l2vpn:l2vpn:
       +--rw vpls-contstraints

     notifications:
       +---n evpn-state-change-notification
          +--ro evpn-instance?   evpn-instance-ref
          +--ro state?           identityref
```

4. YANG Module

The EVPN configuration container is logically divided into

following high level configuration areas:

4.1 Ethernet Segment Yang Module

```
<CODE BEGINS> file "ietf-ethernet-segment@2019-03-09.yang"
module ietf-ethernet-segment {
  namespace "urn:ietf:params:xml:ns:yang:ietf-ethernet-segment";
  prefix "es";

  import ietf-yang-types {
    prefix "yang";
  }

  import ietf-inet-types {
    prefix "inet";
  }

  import ietf-routing-types {
    prefix "rt-types";
  }

  import ietf-interfaces {
    prefix "if";
  }

  import ietf-pseudowires {
    prefix "pw";
  }

  organization  "ietf";
  contact       "ietf";
  description   "ethernet segment";

  revision "2019-03-09" {
    description " - Create an ethernet-segment type and change references " +
                "   to ethernet-segment-identifier " +
                " - Updated Route-target lists to rt-types:vpn-route-targets
" +
                "";
    reference   "";
  }
  revision "2018-02-20" {
    description " - Change the type of attachment circuit to " +
                "   if:interface-ref " +
                "";
    reference   "";
  }

  revision "2017-10-21" {
```

```
      description " - Updated ethernet segment's AC/PW members to " +
                  "   accommodate more than one AC or more than one " +
                  "   PW " +
                  " - Added the new preference based DF election " +
                  "   method " +
                  " - Referenced pseudowires in the new " +
                  "   ietf-pseudowires.yang model " +
                  " - Moved model to NMDA style specified in " +
                  "   draft-dsdt-nmda-guidelines-01.txt " +
                  "";
      reference   "";
    }

    revision "2017-03-08" {
      description " - Updated to use BGP parameters from " +
                  "   ietf-routing-types.yang instead of from " +
                  "   ietf-evpn.yang " +
                  " - Updated ethernet segment's AC/PW members to " +
                  "   accommodate more than one AC or more than one " +
                  "   PW " +
                  " - Added the new preference based DF election " +
                  "   method " +
                  "";
      reference   "";
    }

    revision "2016-07-08" {
      description "   - Added the configuration option to enable or " +
                  "     disable per-EVI/EAD route " +
                  "   - Added PBB parameter backbone-src-mac " +
                  "   - Added operational state branch, initially " +
                  "     to match the configuration branch" +
                  "";
      reference   "";
    }

    revision "2016-06-23" {
      description "WG document adoption";
      reference   "";
    }

    revision "2015-10-15" {
      description "Initial revision";
      reference   "";
    }

    /* Features */
```

```
feature ethernet-segment-bgp-params {
  description "Ethernet segment's BGP parameters";
}

feature ethernet-segment-pbb-params {
  description "Ethernet segment's PBB parameters";
}

/* Typedefs */
typedef status-type {
  type enumeration {
    enum up {
      description "Status is up";
    }
    enum down {
      description "Status is down";
    }
  }
  description "status type";
}

typedef df-election-method-type {
  type enumeration {
    enum default {
      value 0;
      description "The default DF election method";
    }
    enum highest-random-weight {
      value 1;
      description "The highest random weight (HRW) method";
      reference "draft-mohanty-bess-evpn-df-election";
    }
    enum preference {
      value 2;
      description "The preference based method";
      reference "draft-rabadan-bess-evpn-pref-df";
    }
  }
  description "The DF election method type";
}

typedef ethernet-segment-identifier-type {
  type yang:hex-string {
    length "29";
  }
  description "10-octet Ethernet segment identifier (esi),
               ex: 00:5a:5a:5a:5a:5a:5a:5a:5a:5a";
}
```

```
/* EVPN Ethernet Segment YANG Model */

container ethernet-segments {
  description "ethernet-segment";
  list ethernet-segment {
    key "name";
    leaf name {
      type string;
      description "Name of the ethernet segment";
    }
    leaf service-type {
      type string;
      config false;
      description "service-type";
    }
    leaf status {
      type status-type;
      config false;
      description "Ethernet segment status";
    }
    choice ac-or-pw {
      description "ac-or-pw";
      case ac {
        leaf-list ac {
          type if:interface-ref;
          description "Name of attachment circuit";
        }
      }
      case pw {
        leaf-list pw {
          type pw:pseudowire-ref;
          description "Reference to a pseudowire";
        }
      }
    }
    leaf interface-status {
      type status-type;
      config false;
      description "interface status";
    }
    leaf ethernet-segment-identifier {
      type ethernet-segment-identifier-type;
      description "Ethernet segment identifier (esi)";
    }
    choice active-mode {
      mandatory true;
      description "Choice of active mode";
      case single-active {
```

```
            leaf single-active-mode {
              type empty;
              description "single-active-mode";
            }
          }
          case all-active {
            leaf all-active-mode {
              type empty;
              description "all-active-mode";
            }
          }
        }
        container pbb-parameters {
          if-feature ethernet-segment-pbb-params;
          description "PBB configuration";
          leaf backbone-src-mac {
            type yang:mac-address;
            description "backbone-src-mac, only if this is a PBB";
          }
        }
        container bgp-parameters {
          description "BGP parameters";
          container common {
            description "BGP parameters common to all pseudowires";
            list rd-rt {
              if-feature ethernet-segment-bgp-params;
              key "route-distinguisher";
              leaf route-distinguisher {
                type rt-types:route-distinguisher;
                description "Route distinguisher";
              }
              uses rt-types:vpn-route-targets;
              description "A list of route distinguishers and " +
                          "corresponding VPN route targets";
            }
          }
        }
        container df-election {
          description "df-election";
          leaf df-election-method {
            type df-election-method-type;
            description "The DF election method";
          }
          leaf preference {
            when "../df-election-method = 'preference'" {
              description "The preference value is only applicable " +
                          "to the preference based method";
            }
```

```
            type uint16;
            description "The DF preference";
          }
          leaf revertive {
            when "../df-election-method = 'preference'" {
              description "The revertive value is only applicable " +
                          "to the preference method";
            }
            type boolean;
            default true;
            description "The 'preempt' or 'revertive' behavior";
          }
          leaf election-wait-time {
            type uint32;
            description "election-wait-time";
          }
        }
        leaf ead-evi-route {
          type boolean;
          default false;
          description "Enable (true) or disable (false) ead-evi-route";
        }
        leaf esi-label {
          type rt-types:mpls-label;
          config false;
          description "esi-label";
        }
        list member {
          config false;
          leaf ip-address {
            type inet:ip-address;
            description "ip-address";
          }
          description "member of the ethernet segment";
        }
        list df {
          config false;
          leaf service-identifier {
            type uint32;
            description "service-identifier";
          }
          leaf vlan {
            type uint32;
            description "vlan";
          }
          leaf ip-address {
            type inet:ip-address;
            description "ip-address";
```

```
            }
            description "df of an evpn instance's vlan";
          }
          description "An ethernet segment";
        }
      }
    }
    <CODE ENDS>
```

4.2 EVPN Yang Module

```
    <CODE BEGINS> file "ietf-evpn@2019-03-09.yang"
    module ietf-evpn {
      namespace "urn:ietf:params:xml:ns:yang:ietf-evpn";
      prefix "evpn";

      import ietf-inet-types {
        prefix "inet";
      }

      import ietf-yang-types {
        prefix "yang";
      }

      import ietf-routing-types {
        prefix "rt-types";
      }

      import ietf-network-instance {
        prefix "ni";
      }

      import ietf-l2vpn {
        prefix "l2vpn";
      }

      import ietf-pseudowires {
        prefix "pw";
      }

      import ietf-ethernet-segment {
        prefix "es";
      }


      organization  "ietf";
      contact       "ietf";
```

```
     description   "evpn";

     revision "2019-03-09" {
       description " - Incorporated ietf-ethernet-segment model and" +
                   "   normalised ethernet-segment entries on routes " +
                   " - Updated Route-target lists to rt-types:vpn-route-targets
" +
                   "";
       reference   "";
     }

     revision "2018-02-20" {
       description " - Incorporated ietf-network-instance model" +
                   "   on which ietf-l2vpn is now based " +
                   "";
       reference   "";
     }

     revision "2017-10-21" {
       description " - Modified the operational state augment " +
                   " - Renamed evpn-instances-state to evpn-instances" +
                   " - Added vpws-vlan-aware to an EVPN instance " +
                   " - Added a new augment to L2VPN to add EPVN " +
                   " - pseudowire for the case of EVPN VPWS " +
                   " - Added state change notification " +
                   "";
       reference   "";
     }

     revision "2017-03-13" {
       description " - Added an augment to base L2VPN model to " +
                   "   reference an EVPN instance " +
                   " - Reused ietf-routing-types.yang " +
                   "   vpn-route-targets grouping instead of " +
                   "   defining it in this module " +
                   "";
       reference   "";
     }

     revision "2016-07-08" {
       description " - Added operational state" +
                   " - Added a configuration knob to enable/disable " +
                   "   underlay-multicast " +
                   " - Added a configuration knob to enable/disable " +
                   "   flooding of unknonw unicast " +
                   " - Added several configuration knobs " +
                   "   to manage ARP and ND" +
                   "";
       reference   "";
```

```
    }

    revision "2016-06-23" {
      description "WG document adoption";
      reference   "";
    }

    revision "2015-10-15" {
      description "Initial revision";
      reference   "";
    }

    feature evpn-bgp-params {
      description "EVPN's BGP parameters";
    }

    feature evpn-pbb-params {
      description "EVPN's PBB parameters";
    }

    /* Identities */

    identity evpn-notification-state {
      description "The base identity on which EVPN notification " +
                  "states are based";
    }

    identity MAC-duplication-detected {
      base "evpn-notification-state";
      description "MAC duplication is detected";
    }

    identity mass-withdraw-received {
      base "evpn-notification-state";
      description "Mass withdraw received";
    }

    identity static-MAC-move-detected {
      base "evpn-notification-state";
      description "Static MAC move is detected";
    }

    /* Typedefs */

    typedef evpn-instance-ref {
      type leafref {
        path "/evpn/evpn-instances/evpn-instance/name";
      }
```

```
      description "A leafref type to an EVPN instance";
    }

    /* Groupings */

    grouping route-rd-rt-grp {
      description "A grouping for a route's route distinguishers " +
                  "and route targets";
      list rd-rt {
        key "route-distinguisher";
        leaf route-distinguisher {
          type rt-types:route-distinguisher;
          description "Route distinguisher";
        }
        list vpn-target {
          key "route-target";
          leaf route-target {
            type rt-types:route-target;
            description "BGP route target";
          }
          description "A list of route targets";
        }
        description "A list of route distinguishers and " +
                    "corresponding VPN route targets";
      }
    }

    grouping next-hop-label-grp {
      description "next-hop-label-grp";
      leaf next-hop {
        type inet:ip-address;
        description "next-hop";
      }
      leaf label {
        type rt-types:mpls-label;
        description "label";
      }
    }

    grouping next-hop-label2-grp {
      description "next-hop-label2-grp";
      leaf label2 {
        type rt-types:mpls-label;
        description "label2";
      }
    }

    grouping path-detail-grp {
```

```
      description "path-detail-grp";
      container detail {
        config false;
        description "path details";
        container attributes {
          leaf-list extended-community {
            type string;
            description "extended-community";
          }
          description "attributes";
        }
        leaf bestpath {
          type empty;
          description "Indicate this path is the best path";
        }
      }
    }

    /* EVPN YANG Model */

    container evpn {
      description "evpn";
      container common {
        description "common epn attributes";
        choice replication-type {
          description "A choice of replication type";
          case ingress-replication {
            leaf ingress-replication {
              type boolean;
              description "ingress-replication";
            }
          }
          case p2mp-replication {
            leaf p2mp-replication {
              type boolean;
              description "p2mp-replication";
            }
          }
        }
      }
      container evpn-instances {
        description "evpn-instances";
        list evpn-instance {
          key "name";
          description "An EVPN instance";
          leaf name {
            type string;
            description "Name of EVPN instance";
```

```
          }
          leaf evi {
            type uint32;
            description "evi";
          }
          container pbb-parameters {
            if-feature "evpn-pbb-params";
            description "PBB parameters";
            leaf source-bmac {
              type yang:hex-string;
              description "source-bmac";
            }
          }
          container bgp-parameters {
            description "BGP parameters";
            container common {
              description "BGP parameters common to all pseudowires";
              list rd-rt {
                if-feature evpn-bgp-params;
                key "route-distinguisher";
                leaf route-distinguisher {
                  type rt-types:route-distinguisher;
                  description "Route distinguisher";
                }
                uses rt-types:vpn-route-targets;
                description "A list of route distinguishers and " +
                            "corresponding VPN route targets";
              }
            }
          }
          leaf arp-proxy {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) ARP proxy";
          }
          leaf arp-suppression {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) " +
                        "ARP suppression";
          }
          leaf nd-proxy {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) ND proxy";
          }
          leaf nd-suppression {
            type boolean;
```

```
            default false;
            description "Enable (TRUE) or disable (FALSE) " +
                        "ND suppression";
          }
          leaf underlay-multicast {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) " +
                        "underlay multicast";
          }
          leaf flood-unknown-unicast-supression {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) " +
                        "flood unknown unicast suppression";
          }
          leaf vpws-vlan-aware {
            type boolean;
            default false;
            description "Enable (TRUE) or disable (FALSE) " +
                        "VPWS VLAN aware";
          }
          container routes {
            config false;
            description "routes";
            list ethernet-auto-discovery-route {
              uses route-rd-rt-grp;
              leaf ethernet-segment-identifier {
                type es:ethernet-segment-identifier-type;
                description "Ethernet segment identifier (esi)";
              }
              leaf ethernet-tag {
                type uint32;
                description "An ethernet tag (etag) indentifying a " +
                            "broadcast domain";
              }
              list path {
                uses next-hop-label-grp;
                uses path-detail-grp;
                description "path";
              }
              description "ethernet-auto-discovery-route";
            }
            list mac-ip-advertisement-route {
              uses route-rd-rt-grp;
              leaf ethernet-segment-identifier {
                type es:ethernet-segment-identifier-type;
                description "Ethernet segment identifier (esi)";
```

```
            }
            leaf ethernet-tag {
              type uint32;
              description "An ethernet tag (etag) indentifying a " +
                          "broadcast domain";
            }
            leaf mac-address {
              type yang:mac-address;
              description "Route mac address";
            }
            leaf mac-address-length {
              type uint8 {
                range "0..48";
              }
              description "mac address length";
            }
            leaf ip-prefix {
              type inet:ip-prefix;
              description "ip-prefix";
            }
            list path {
              uses next-hop-label-grp;
              uses next-hop-label2-grp;
              uses path-detail-grp;
              description "path";
            }
            description "mac-ip-advertisement-route";
          }
          list inclusive-multicast-ethernet-tag-route {
            uses route-rd-rt-grp;
            leaf originator-ip-prefix {
              type inet:ip-prefix;
              description "originator-ip-prefix";
            }
            list path {
              uses next-hop-label-grp;
              uses path-detail-grp;
              description "path";
            }
            description "inclusive-multicast-ethernet-tag-route";
          }
          list ethernet-segment-route {
            uses route-rd-rt-grp;
            leaf ethernet-segment-identifier {
              type es:ethernet-segment-identifier-type;
              description "Ethernet segment identifier (esi)";
            }
            leaf originator-ip-prefix {
```

```
            type inet:ip-prefix;
            description "originator ip-prefix";
          }
          list path {
            leaf next-hop {
              type inet:ip-address;
              description "next-hop";
            }
            uses path-detail-grp;
            description "path";
          }
          description "ethernet-segment-route";
        }
        list ip-prefix-route {
          uses route-rd-rt-grp;
          leaf ethernet-segment-identifier {
            type es:ethernet-segment-identifier-type;
            description "Ethernet segment identifier (esi)";
          }
          leaf ip-prefix {
            type inet:ip-prefix;
            description "ip-prefix";
          }
          list path {
            uses next-hop-label-grp;
            uses path-detail-grp;
            description "path";
          }
          description "ip-prefix route";
        }
      }
      container statistics {
        config false;
        description "Statistics";
        leaf tx-count {
          type yang:zero-based-counter32;
          description "transmission count";
        }
        leaf rx-count {
          type yang:zero-based-counter32;
          description "receive count";
        }
        container detail {
          description "Detailed statistics";
          leaf broadcast-tx-count {
            type yang:zero-based-counter32;
            description "broadcast transmission count";
          }
```

```
            leaf broadcast-rx-count {
              type yang:zero-based-counter32;
              description "broadcast receive count";
            }
            leaf multicast-tx-count {
              type yang:zero-based-counter32;
              description "multicast transmission count";
            }
            leaf multicast-rx-count {
              type yang:zero-based-counter32;
              description "multicast receive count";
            }
            leaf unknown-unicast-tx-count {
              type yang:zero-based-counter32;
              description "unknown unicast transmission count";
            }
            leaf unknown-unicast-rx-count {
              type yang:zero-based-counter32;
              description "unknown-unicast receive count";
            }
          }
        }
      }
    }

    /* augments */

    augment "/pw:pseudowires/pw:pseudowire/pw:pw-type" {
      description "Augment for an L2VPN instance to add EVPN VPWS " +
                  "pseudowire";
      case evpn-pw {
        container evpn-pw {
          description "EVPN pseudowire";
          leaf remote-id {
            type uint32;
            description "Remote pseudowire ID";
          }
          leaf local-id {
            type uint32;
            description "Local pseudowire ID";
          }
        }
      }
    }

    augment "/ni:network-instances/ni:network-instance/ni:ni-type" +
            "/l2vpn:l2vpn" {
```

```
      description "Augment for an L2VPN instance and EVPN association";
      leaf evpn-instance {
        type evpn-instance-ref;
        description "Reference to an EVPN instance";
      }
    }

    augment "/ni:network-instances/ni:network-instance/ni:ni-type" +
            "/l2vpn:l2vpn" {
      when "l2vpn:type = 'l2vpn:vpls-instance-type'" {
        description "Constraints only for VPLS pseudowires";
      }
      description "Augment for VPLS instance";
      container vpls-contstraints {
        must "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:pw/l2vpn:name]" +
             "             /evpn-pw/remote-id)) and " +
             "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:pw/l2vpn:name]" +
             "             /evpn-pw/local-id)) and " +
             "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:primary-pw/l2vpn:name]" +
             "             /evpn-pw/remote-id)) and " +
             "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:primary-pw/l2vpn:name]" +
             "             /evpn-pw/local-id)) and " +
             "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:backup-pw/l2vpn:name]" +
             "             /evpn-pw/remote-id)) and " +
             "not(boolean(/pw:pseudowires/pw:pseudowire" +
             "                [pw:name = current()/../l2vpn:endpoint" +
             "             /l2vpn:backup-pw/l2vpn:name]" +
             "             /evpn-pw/local-id))" {
          description "A VPLS pseudowire must not be EVPN PW";
        }
        description "VPLS constraints";
      }
    }

    /* Notifications */

    notification evpn-state-change-notification {
      description "EVPN state change notification";
```

```
      leaf evpn-instance {
        type evpn-instance-ref;
        description "Related EVPN instance";
      }
      leaf state {
        type identityref {
          base evpn-notification-state;
        }
        description "State change notification";
      }
    }
  }
  <CODE ENDS>
```

5. Security Considerations

The configuration, state, action and notification data defined in this document are designed to be accessed via the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the secure transport layer and the mandatory-to-implement secure transport is SSH [RFC6242]. The NETCONF access control model [RFC6536] provides means to restrict access for particular NETCONF users to a pre-configured subset of all available NETCONF protocol operations and content.

The security concerns listed above are, however, no different than faced by other routing protocols. Hence, this draft does not change any underlying security issues inherent in [I-D.ietf-netmod-routing-cfg]

6. IANA Considerations

None.

7. References

7.1. Normative References

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

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC8294] Liu, X., Qu, Y., Lindem, A., Hopps, C., and L. Berger, "Common YANG Data Types for the Routing Area", RFC 8294,

DOI 10.17487/RFC8294, December 2017, <https://www.rfc-editor.org/info/rfc8294>.

7.2. Informative References

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <https://www.rfc-editor.org/info/rfc6536>.

[RFC7209] Sajassi, A., Aggarwal, R., Uttaro, J., Bitar, N., Henderickx, W., and A. Isaac, "Requirements for Ethernet VPN (EVPN)", RFC 7209, DOI 10.17487/RFC7209, May 2014, <https://www.rfc-editor.org/info/rfc7209>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

Authors' Addresses

Patrice Brissette
Cisco Systems, Inc.
EMail: pbrisset@cisco.com

Himanshu Shah
Ciena Corporation
EMail: hshah@ciena.com

Helen Chen
Jabil
EMail: Ing-Wher_Chen@jabil.com

Iftekar Hussain
Infinera Corporation
EMail: ihussain@infinera.com

Kishore Tiruveedhula
Juniper Networks
EMail: kishoret@juniper.net

Jorge Rabadan
Nokia
EMail: jorge.rabadan@nokia.com

Ali Sajassi
Cisco Systems, Inc.
EMail: sajassi@cisco.com

Zhenbin Li
Huawei Technologies
EMail: lizhenbin@huawei.com

BESS Working Group

K. Patel
Arrcus

Internet Draft

A. Sajassi
Cisco

Category: Standards Track

J. Drake
Z. Zhang
Juniper Networks

W. Henderickx
Nokia

Expires: May 22, 2019

October 22, 2018

Virtual Hub-and-Spoke in BGP EVPNs
draft-keyupate-bess-evpn-virtual-hub-01

Abstract

Ethernet Virtual Private Network (EVPN) solution is becoming pervasive for Network Virtualization Overlay (NVO) services in data center (DC) applications and as the next generation virtual private LAN services in service provider (SP) applications.

The use of host IP default route and host unknown MAC route within a DC is well understood in order to ensure that leaf nodes within a DC only learn and store host MAC and IP addresses for that DC. All other host MAC and IP addresses from remote DCs are learned and stored in DC GW nodes thus alleviating leaf nodes from learning host MAC and IP addresses from the remote DCs.

This draft further optimizes the MAC and IP address learning at the leaf nodes such that a leaf node within a DC only needs to learn and store MAC and IP addresses associated with the sites directly connected to it. A leaf node does not need to learn and store MAC and IP addresses from any other leaf nodes thus reducing the number of learned MACs and IP addresses per EVI substantially.

The modifications provided by this draft updates and extends RFC7024 for BGP EVPN Address Family.

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/1id-abstracts.html

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

Table of Contents

1. Introduction

Ethernet Virtual Private Network (EVPN) solution is becoming pervasive for Network Virtualization Overlay (NVO) services in data center (DC) applications and as the next generation virtual private LAN services in service provider (SP) applications.

With EVPN, providing any-to-any connectivity among sites of a given EVPN Instance (EVI) would require each Provider Edge (PE) router connected to one or more of these sites to hold all the host MAC and IP addresses for that EVI. The use of host IP default route and host unknown MAC route within a DC is well understood in order to alleviate the learning of host MAC and IP addresses to only leaf nodes (PEs) within that DC. All other host MAC and IP addresses from remote DCs are learned and stored in DC GW nodes thus alleviating leaf nodes from learning host MAC and IP addresses from the remote DCs.

This draft further optimizes the MAC and IP address learning at the leaf nodes such that a leaf node within a DC only needs to learn and store MAC and IP addresses associated with the sites directly connected to it. A leaf node does not need to learn and store MAC and IP addresses from any other leaf nodes thus reducing the number of learned MACs and IP addresses per EVI substantially.

[RFC7024] provides rules for Hub and Spoke VPNs for BGP L3VPNs. This draft updates and extends [RFC7024] for BGP EVPN Address Family. This draft provides rules for Originating and Processing of the EVPN host unknown MAC route and host default IP route by EVPN Virtual Hub (V-HUB). This draft also provides rules for the handling of the BUM traffic in Hub and Spoke EVPNs and handling of ARP suppression.

The leaf nodes and DC GW nodes in a data center are referred to as Virtual Spokes (V-spokes) and Virtual Hubs (V-hubs) respectively. A set of V-spoke can be associated with one or more V-hubs. If a V-spokes is associated with more than one V-hubs, then it can load balanced traffic among these V-hubs. Different V-spokes can be associated with different sets of V-hubs such that at one extreme each V-spoke can have a different V-hub set although this may not be desirable and a more typical scenario may be to associate a set of V-spokes to a set of V-hubs - e.g., topology for a DC POD where a set of V-spokes are associated with a set of spine nodes or DC GW nodes.

In order to avoid repeating many of the materials covered in [RFC7024], this draft is written as a delta document with its sections organized to follow those of that RFC with only delta description pertinent to EVPN operation in each section. Therefore, it is assumed that the readers are very familiar with [RFC7024] and

EVPN.

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. Terminology

ARP: Address Resolution Protocol
BEB: Backbone Edge Bridge
B-MAC: Backbone MAC Address
CE: Customer Edge
C-MAC: Customer/Client MAC Address
ES: Ethernet Segment
ESI: Ethernet Segment Identifier
IRB: Integrated Routing and Bridging
LSP: Label Switched Path
MP2MP: Multipoint to Multipoint
MP2P: Multipoint to Point
ND: Neighbor Discovery
NA: Neighbor Advertisement
P2MP: Point to Multipoint
P2P: Point to Point
PE: Provider Edge
EVPN: Ethernet VPN
EVI: EVPN Instance
RT: Route Target

Single-Active Redundancy Mode: When only a single PE, among a group of PEs attached to an Ethernet segment, is allowed to forward traffic to/from that Ethernet Segment, then the Ethernet segment is defined to be operating in Single-Active redundancy mode.

All-Active Redundancy Mode: When all PEs attached to an Ethernet segment are allowed to forward traffic to/from that Ethernet Segment, then the Ethernet segment is defined to be operating in All-Active redundancy mode.

4. Routing Information Exchange for EVPN routes

[RFC7432] defines multiple Route Types NLRI along with procedures for

advertisements and processing of these routes. Some of these procedures are impacted as the result of hub-and-spoke architecture. The routing information exchange among the hub, spoke, and vanilla PEs are subject to the same rules as described in section 3 of [RFC7024]. Furthermore, if there are any changes to the EVPN route advisements and processing from that of [RFC7432], they are described below.

5. EVPN unknown MAC route

Section 3 of [RFC7024] talks about how a V-hub of a given VPN must export a VPN-IP default route for that VPN and this route must be exported to only the V-spokes of that VPN associated with that V-hub. [DCI-EVPN] defines the notion of the unknown MAC route for an EVI which is analogous to a VPN-IP default route for a VPN. This unknown MAC route is exported by a V-hub to its associated V-spokes. If multiple V-hubs are associated with a set of V-spokes, then each V-hub advertises it with a distinct RD when originating this route. If a V-spoke imports several of these unknown MAC routes and they all have the same preference, then traffic from the V-spoke to other sites of that EVI would be load balanced among the V-hubs.

5.1. Originating EVPN Unknown MAC Route by a V-Hub

Section 7.3 of the [RFC7024] defines procedures for originating a VPN-IP default route for a VPN. The same procuedures apply when a V-hub wants to originate EVPN unknown MAC route for a given EVI. The V-hub MUST announce unknown MAC route using the MAC/IP advertisement route along with the Default Gateway extended community as defined in section 10.1 of the [RFC7432].

5.2. Processing VPN-MAC EVPN unknown Route by a V-SPOKE

Within a given EVPN, a V-spoke MUST import all the unknown MAC routes unless the route-target mismatch happens. The processing of the received VPN-MAC EVPN default route follows the rules explained in the section 3 of the [RFC7024]. The unknown MAC route MUST be installed according to the rules of MAC/IP Advertisement route installation rules in section 9.2.2 of [RFC7024].

In absense of any more specific VPN-MAC EVPN routes, V-spokes installing the unknown MAC route MUST use the route when performing ARP proxy. This behavior would allow V-Spokes to forward the traffic towards V-Hub.

5.3. Aliasing

[RFC7432] describes the concept and procedures for Aliasing where a station is multi-homed to multiple PEs operating in an All-Active redundancy mode, it is possible that only a single PE learns a set of MAC addresses associated with traffic transmitted by the station. [RFC7432] describes the concepts and procedures for Aliasing, which occurs when a CE is multi-homed to multiple PE nodes, operating in all-active redundancy mode, but not all of the PEs learn the CE's set of MAC addresses. This leads to a situation where remote PEs receive MAC advertisement routes, for these addresses, from a single NVE even though multiple NVEs are connected to the multi-homed station. As a result, the remote NVEs are not able to effectively load-balance traffic among the NVEs connected to the multi-homed Ethernet segment.

To alleviate this issue, EVPN introduces the concept of Aliasing. This refers to the ability of a PE to signal that it has reachability to a given locally attached Ethernet segment, even when it has learnt no MAC addresses from that segment. The Ethernet A-D per-EVI route is used to that end. Remote PEs which receive MAC advertisement routes with non-zero ESI SHOULD consider the MAC address as reachable via all NVEs that advertise reachability to the relevant Segment using Ethernet A-D routes with the same ESI and with the Single-Active flag reset.

This procedure is impacted for virtual hub-and-spoke topology because a given V-spoke does not receive any MAC/IP advertisements from remote V-spokes; therefore, there is no point in propagating Ethernet A-D per-EVI route to the remote V-spokes. In this solution, the V-hubs terminate the Ethernet A-D per-EVI route (used for Aliasing) and follows the procedures described in [RFC7432] for handling this route.

There are scenarios for which it is desirable to establish direct communication path between a pair of V-spokes for a given host MAC address. In such scenario, the advertising V-spoke advertises both the MAC/IP route and Ethernet A-D per-EVI route with the RT of V-hub (RT-VH) per section 3 of [RFC7024]. The use of RT-VH, ensures that these routes are received by the V-spokes associated with that V-hub set and thus enables the V-spokes to perform the Aliasing procedure.

In summary, PE devices (V-hubs in general and V-spokes occasionally) that receive EVPN MAC/IP route advertisements (associated with a multi-homed site) need to also receive the associated Ethernet A-D per-EVI route advertisement(s) in order for them to perform Aliasing procedure.

5.4. Split-Horizon & Mass Withdraw

[RFC7432] uses Ethernet A-D per-ES route to a) signal to remote PEs the multi-homing redundancy type (Single-Active versus All-Active), b) advertise ESI label for split-horizon filtering when MPLS encapsulation is used, and c) advertise mass-withdraw when a failure of an access interface impacts many MAC addresses. This route does not need to be advertise from a V-spoke to any remote V-spoke unless a direct communication path between a pair of spoke is needed for a given flow.

Even if communication between a pair of V-spoke is needed for just a single flow, the Ethernet A-D per ES route needs to be advertised from the originating V-spoke for that ES which may handle tens or hundreds of thousands of flows. This is because in order to perform Aliasing function for a given flow, the Ethernet A-D per-EVI route is needed and this route itself is dependent on the Ethernet A-D per-ES route. In such scenario, the advertising V-spoke advertises the Ethernet A-D per-ES route with the RT of V-hub (RT-VH) per section 3 of [RFC7024].

In summary, PE devices (V-hubs in general and V-spokes occasionally) that receive EVPN MAC/IP route advertisements (associated with a multi-homed site) need to also receive the associated Ethernet A-D per-ES route advertisement(s).

6. Forwarding Considerations

6.1. IP-only Forwarding

When EVPN operates in IP-only forwarding mode using EVPN Route Type 5, then all forwarding considerations in section 4 of [RFC7024] are directly applicable here.

6.2. MAC-only Forwarding - Bridging

When EVPN operates in MAC-only forwarding mode (i.e., bridging mode), then for a given EVI, the MPLS label that a V-hub advertises with anUnknown MAC address MUST be the label that identifies the MAC-VRF of the V-hub in absense of a more specific MAC route. When the V-hub receives a packet with such label, the V- hub pops the label and determines further disposition of the packet based on the lookup in the MAC-VRF. Otherwise, the MPLS label of the matching more specific route is used and packet is is forwarded towards the associated NEXTHOP of the more specific route.

6.3. MAC and IP Forwarding - IRB

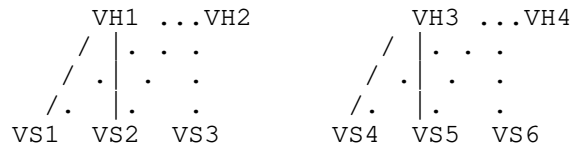
When a EVPN speaker operates in IRB mode, it implements both the IP and MAC forwarding Modes (aka Integrated Routing and Bridging - IRB). On a packet by packet basis, the V-spoke decides whether to do forwarding based on a MAC address lookup (bridge) or based on a IP address lookup (route). If the host destination MAC address is that of the IRB interface (i.e., if the traffic is inter-subnet), then the V-spoke performs an additional IP lookup in the IP-VRF. However, if the host destination MAC address is that of an actual host MAC address (i.e., the traffic is intra- subnet) , then the V-spoke only performs a MAC lookup in the MAC-VRF. The procedure specified in Section 6.1 and Section 6.2 are applicable to inter-subnet and intra-subnet forwarding respectively. For intra- subnet traffic, if the MAC address is not found in the MAC-VRF, then the V-spoke forwards the traffic to the V-hub with the MPLS label received from the V-hub for the unknown MAC address. For the Inter- subnet traffic, if the IP prefix is not found in the IP-VRF, then the V-spoke forwards the traffic to the V-hub with the MPLS label received from the V-hub for the default IP address.

7. Handling of Broadcast and Multicast traffic

Just like that V-spoke to V-spoke known unicast traffic is relayed by V-hubs, V-spoke to V-spoke BUM traffic can also relayed by V-hubs. This is especially desired if Ingress Replication (IR) would be used otherwise for V-spokes to send traffic to other V-spokes. This way, a V-spoke can unicast BUM traffic to a single V-hub, who will then relay the traffic. This achieves Assisted Replication, and reduces multicast state in the core. Note that a V-hub may relay traffic using MPLS P2MP tunnels or BIER as well as IR. While a V-spoke may use P2MP tunnels or BIER to send traffic to V-hubs, this specification focuses on using IR by V-spokes.

In this particular section, all traffic refers to BUM traffic unless explicitly stated otherwise. The term PE refers to a V-hub or V-spoke when there is no need to distinguish the two.

Consider the following topology, where V-spokes VS1/2/3 are associated with V-hubs VH1/2 in one cluster, and V-spokes VS4/5/6 are associated with V-hubs VH3/4 in another cluster. Note that the lines/dots in the diagram indcate association, not connection.



7.1. Split Horizon

When VH1 relays traffic that it receives from VS1, in case of IR it MUST not send traffic back to VS1, and in case of P2MP tunnel it must indicate that traffic is sourced from VS1 so that VS1 will discard the traffic. In case of IR with IP unicsat tunnels, the outer source IP address identifies the sending PE. In case of IR with MPLS unicast tunnels, VH1 must advertise different labels to different PEs, so that it can identify the sending PE based on the label in the traffic from a V-spoke.

If MPLS P2MP/multicast tunnels (including VXLAN-GPE and MPLS-over-GRE/UDP) are used by a V-hub to relay traffic, an upstream allocated (by the V-hub) label MUST be imposed in the label stack to identify the source of the V-spoke. The label is advertised as part of the PE Distinguisher (PED) Label Attribute of the Inclusive Multicast Ethernet Tag (IMET) route from the V-hub, as specified in Section 8 of [RFC 6514].

Notice that an "upstream-assigned" label used by a V-hub to send traffic with on a P2MP tunnel to identify the source V-spoke is the same "downstream-assigned" label used by the V-hub to receive traffic on the IR tunnel from the V-spoke. Therefore, the same PED Label attribute serves two purposes. With [RFC 6514], a PED label may only identify a PE but not a particular VPN. Here the PED label identifies both the PE and a particular EVI/BD. A V-spoke programs its context MPLS forwarding table for the V-hub to discard any traffic with the PED label that the V-hub advertised for this V-spoke, or pop other PED labels and direct traffic into a corresponding EVI for L2 forwarding.

Note that a V-hub cannot use VXLAN/NVGRE multicast tunnels to relay traffic because if the V-hub uses the source V-spoke's IP address in the outer IP header (for the purpose of identifying the source V-spoke), multicast RPF would fail and the packets will be discarded.

7.2. Route Advertisement

As with other route types, IMET routes from V-hubs are advertised with RT-VH and RT-EVI so they are imported by associated V-spokes and all V-hubs. They carry the PED Label attribute as described above.

IMET routes from V-spokes are advertised with RT-EVI so they are imported by all V-hubs. They also carry PED Label attribute for multi-homing split horizon purpose if and only if V-hubs uses IR to relay traffic.

If a V-hub uses RSVP-TE P2MP tunnel, IR, or BIER to send or relay traffic, all other PEs (V-hubs or V-spokes) will receive traffic directly because the V-hub sees all PEs. If a V-hub uses mLDP P2MP tunnel to send or relay traffic, only its associated V-spokes and all V-hubs will see the V-hub's IMET route and join the tunnel announced in the route. Another V-hub need to relay traffic to its associated V-spokes that are not associated with this V-hub.

For that V-hub to announce the mLDP relay tunnel in its cluster, it needs to advertise a (*,*) S-PMSI AD route, as specified in [BUM-PROCEDURE]. The route is advertised with the RT-VH for that cluster, and associated V-spokes will join the tunnel announced in the S-SPMI AD route.

7.3. Designated Forwarder in a Cluster

When there are multiple V-hubs in a cluster, a V-spoke in that cluster decides by itself to which V-hub to send traffic. If the receiving V-hub uses mLDP tunnel to relay traffic, V-hubs in other clusters need to further relay traffic, but only one V-hub in each cluster can do so. As a result, a DF must be elected among the V-hubs for each cluster.

The election is similar to DF election in RFC 7432, with the folllowing differences.

o Instead of using Ethernet Segment route to discover the PEs on a multi-homing ES, the IMET route are used to determine the V-hubs in the same cluster - they all carry the same pair of RT-EVI and RT-VH, and advertises the unknown mac route.

o Instead of using VLAN to do per-VLAN DF election, the Local Administration Field of the RT-EVI is used to do per-EVI DF election.

7.4. Traffic Forwarding Rules

When a PE needs to forward received traffic from local Attachment Circuits (ACs) or remote PEs to local ACs, it follows the rules in RFC 7432, except that traffic sourced from this local PE but relayed

back on a p2mp tunnel is discarded. It may also need to forward to other PEs, subject to rules in the following sections.

7.4.1. Traffic from Local ACs

Traffic from a V-hub's local ACs is forwarded using the tunnel announced in its IMET route, as specified in RFC 7432. In case of an mLDP tunnel, the traffic need to be relayed by V-hubs of other clusters to their associated V-spokes. For other tunnel types, no relay is needed.

Traffic from a V-spoke's local ACs is forwarded to an associated V-hub of its choice. In case of MPLS IR, the label in the V-hub's IMET route's PED attribute corresponding to this V-spoke is used.

7.4.2. Traffic Received by a V-hub from Another PE

When a V-hub receives traffic from an associated V-spoke, it needs to relay to other PEs, using the tunnel announced in its IMET route. In case of IR or BIER, the source V-spoke, which is determined from the incoming label or source IP address, is excluded from the replication list. In case of a P2MP tunnel, the popped incoming label is imposed again to identify the source PE, before the tunnel label is imposed.

When a V-hub receives traffic from another V-hub on a P2MP tunnel, and the tunnel is announced in an IMET route carrying the same RT-VH as this V-hub is configured with, it does not need to relay the traffic. Otherwise, the traffic is from a V-hub in a different cluster, and this V-hub needs to relay to its associated V-spokes, if and only if it is the DF for this cluster, using the tunnel announced in its (*,*) S-PMSI route carrying its RT-VH.

When a V-hub recevies traffic from another V-hub via IR or BIER, it does not further relay the traffic as that V-hub can reach all PEs.

7.4.3. Traffic received by a V-spoke from a V-hub

In case of P2MP tunnel, the V-spoke discards the traffic if the label following the tunnel label identifies the V-spoke itself.

7.5. Multi-homing support

Consider that an ES spans across two V-spokes in the same cluster and the V-hub uses MPLS IR to relay traffic. With ESI Label split horizon method, a source V-spoke uses the ESI label advertised by the V-hub for the ES, and the V-hub must change that to the ESI label advertised by receiving v-spokes when it relays traffic. That means V-hubs must advertise ESI labels for all multi-homing segments, even

when they're not on those segments. They must also do double label swap (EVI/BD label and ESI label) or mac lookup when relaying traffic.

There are two methods detailed below to avoid that complexity. Either one MAY be used.

7.5.1 Domain-wide Common Block (DCB) Label

[draft-zzhang-bess-mvpn-evpn-aggregation-label] proposes for all PEs on an MHES to use the same ESI label allocated from a Domain-wide Common Block. Not only does that have the advantages described in that document, but also It avoids the MHES complexity with Virtual Hub and Spoke as mentioned above, because the V-Hubs do not need to care about the ESI label at all any more.

7.5.2 Local Bias

If DCB labels cannot be used, then Local Bias can be used even For EVPN MPLS. The PED label following the mpls transport tunnel label or BIER header identifies the PE that originated the traffic in addition to identifying the EVI/BD.

If a V-hub uses P2MP or BIER to relay traffic, the PED label is one of the labels in the PE Distinguiser Label attribute in the V-hub's IMET route, allocated by the V-hub for the source V-spoke.

If a V-hub uses IR to relay traffic, for each V-spoke that it relays to, the PED label advertised by that receiving V-spoke for the source V-spoke needs to be imposed by the V-hub. For that purpose, each V-spoke must include the PED Label attribute in its IMET route, to advertise different labels for different PEs. It discovers the PEs that it needs to advertise labels for via the PED label Attribute in the V-hub's IMET route.

7.6. Direct V-spoke to V-spoke traffic

It may be desired for allow direct V-spoke to V-spoke traffic in a cluster, without the relay by a V-hub. To do that, V-spokes advertise their IMET routes with both RT-VH and RT-EVI. Forwarding rules will be specified in future revisions.

8. ARP/ND Suppression

[RFC7432] defines the procedures for ARP/ND suppression where a PE can terminate gratuitous ARP/ND request message from directly connected site and advertises the associated MAC and IP addresses in an EVPN MAC/IP advertisement route to all other remote PEs. The

remote PEs that receive this EVPN route advertisement, install the MAC/IP pair in their ARP/ND cache table thus enabling them to terminate ARP/ND requests and generate ARP/ND responses locally thus suppressing the flooding of ARP/ND requests over the EVPN network.

In this hub-and-spoke approach, the ARP suppression needs to be performed by both the EVPN V-hubs as well V-spokes as follow. When a V-Spoke receives a gratuitous ARP/ND request, it terminates it and stores the source MAC/IP pair in its ARP/ND cache table. Then, it advertises the source MAC/IP pair to its associated V-Hubs using EVPN MAC/IP advertisement route. The V-Hubs upon receiving this EVPN route advertisement, create an entry in their ARP/ND cache table for this MAC/IP pair.

Now when a V-Spoke receives an ARP/ND request, it first looks up its ARP cache table, if an entry for that MAC/IP pair is found, then an ARP/ND response is generated locally and sent to the CE. However, if an entry is not found, then the ARP/ND request is unicasted to one of the V-hub associated with this V-spoke. Since, the associated V-hub keeps all the MAC/IP ARP entries in its cache table, it can formulate and ARP/ND response and forward it to that CE via the corresponding V-spoke.

9. IANA Considerations

There is no additional IANA considerations for PBB-EVPN beyond what is already described in [RFC7432].

10. Security Considerations

All the security considerations in [RFC7432] apply directly to this document because this document leverages [RFC7432] control plane and their associated procedures - although not the complete set but rather a subset.

This draft does not introduce any new security considerations beyond that of [RFC7432] and [RFC4761] because advertisements and processing of B-MAC addresses follow that of [RFC7432], and processing of C-MAC addresses follow that of [RFC4761] - i.e, B-MAC addresses are learned in control plane and C-MAC addresses are learned in data plane.

11. Acknowledgements

The authors would like to thank Yakov Rekhter for initial idea discussions.

12. Change Log

Initial Version: Sep 21 2014 Original Name: draft-keyupate-evpn-virtual-hub-00.txt

13. References

13.1. Normative References

[RFC7024] Jeng, H., Uttaro, J., Jalil, L., Decraene, B., Rekhter, Y., and R. Aggarwal, "Virtual Hub-and-Spoke in BGP/MPLS VPNs", RFC 7024, October 2013.

[RFC7432] A. Sajassi, et al., "BGP MPLS Based Ethernet VPN", RFC 7432 , February 2015.

13.2. Informative References

[RFC7080] A. Sajassi, et al., "Virtual Private LAN Service (VPLS) Interoperability with Provider Backbone Bridges", RFC 7080, December 2013.

[RFC7209] D. Thaler, et al., "Requirements for Ethernet VPN (EVPN)", RFC 7209, May 2014.

[RFC4389] A. Sajassi, et al., "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, April 2006.

[RFC4761] K. Kompella, et al., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, Jauary 2007.

[OVERLAY] A. Sajassi, et al., "A Network Virtualization Overlay Solution using EVPN", draft-ietf-bess-evpn-overlay-01, work in progress, February 2015.

14. Authors' Addresses

Keyur Patel
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134, US
Email: keyupate@cisco.com

Ali Sajassi

Cisco
170 West Tasman Drive
San Jose, CA 95134, US
Email: sajassi@cisco.com

Yakov Rekhter
Juniper Networks, Inc.
Email: yakov@juniper.net

John E. Drake
Juniper Networks, Inc.
Email: jdrake@juniper.net

Zhaohui Zhang
Juniper Networks, Inc.
Email: zzhang@juniper.net

Wim Henderickx
Nokia
Email: wim.henderickx@nokia.com

BESS Working Group
K. Patel
Arrcus
Internet Draft
A. Sajassi
Category: Standards Track
Cisco
J. Drake
Z. Zhang
Juniper Networks
W. Henderickx
Nokia

Expires: March 02, 2020
September 02, 2019

Virtual Hub-and-Spoke in BGP EVPNs
draft-keyupate-bess-evpn-virtual-hub-02

Abstract

Ethernet Virtual Private Network (EVPN) solution is becoming pervasive for Network Virtualization Overlay (NVO) services in data center (DC) applications and as the next generation virtual private LAN services in service provider (SP) applications.

The use of host IP default route and host unknown MAC route within a DC is well understood in order to ensure that leaf nodes within a DC only learn and store host MAC and IP addresses for that DC. All other host MAC and IP addresses from remote DCs are learned and stored in DC GW nodes thus alleviating leaf nodes from learning host MAC and IP addresses from the remote DCs.

This draft further optimizes the MAC and IP address learning at the leaf nodes such that a leaf node within a DC only needs to learn and store MAC and IP addresses associated with the sites directly connected to it. A leaf node does not need to learn and store MAC and IP addresses from any other leaf nodes thus reducing the number of learned MACs and IP addresses per EVI substantially.

The modifications provided by this draft updates and extends RFC7024 for BGP EVPN Address Family.

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/1id-abstracts.html

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

Table of Contents

1. Introduction

Ethernet Virtual Private Network (EVPN) solution is becoming pervasive for Network Virtualization Overlay (NVO) services in data center (DC) applications and as the next generation virtual private LAN services in service provider (SP) applications.

With EVPN, providing any-to-any connectivity among sites of a given EVPN Instance (EVI) would require each Provider Edge (PE) router connected to one or more of these sites to hold all the host MAC and IP addresses for that EVI. The use of host IP default route and host unknown MAC route within a DC is well understood in order to alleviate the learning of host MAC and IP addresses to only leaf nodes (PEs) within that DC. All other host MAC and IP addresses from remote DCs are learned and stored in DC GW nodes thus alleviating leaf nodes from learning host MAC and IP addresses from the remote DCs.

This draft further optimizes the MAC and IP address learning at the leaf nodes such that a leaf node within a DC only needs to learn and store MAC and IP addresses associated with the sites directly connected to it. A leaf node does not need to learn and store MAC and IP addresses from any other leaf nodes thus reducing the number of learned MACs and IP addresses per EVI substantially.

[RFC7024] provides rules for Hub and Spoke VPNs for BGP L3VPNs. This draft updates and extends [RFC7024] for BGP EVPN Address Family. This draft provides rules for Originating and Processing of the EVPN host unknown MAC route and host default IP route by EVPN Virtual Hub (V-HUB). This draft also provides rules for the handling of the BUM traffic in Hub and Spoke EVPNs and handling of ARP suppression.

The leaf nodes and DC GW nodes in a data center are referred to as Virtual Spokes (V-spokes) and Virtual Hubs (V-hubs) respectively. A set of V-spoke can be associated with one or more V-hubs. If a V-spokes is associated with more than one V-hubs, then it can load balanced traffic among these V-hubs. Different V-spokes can be associated with different sets of V-hubs such that at one extreme each V-spoke can have a different V-hub set although this may not be desirable and a more typical scenario may be to associate a set of V-spokes to a set of V-hubs - e.g., topology for a DC POD where a set of V-spokes are associated with a set of spine nodes or DC GW nodes.

In order to avoid repeating many of the materials covered in [RFC7024], this draft is written as a delta document with its sections organized to follow those of that RFC with only delta description pertinent to EVPN operation in each section. Therefore, it is assumed that the readers are very familiar with [RFC7024] and

EVPN.

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. Terminology

ARP: Address Resolution Protocol
BEB: Backbone Edge Bridge
B-MAC: Backbone MAC Address
CE: Customer Edge
C-MAC: Customer/Client MAC Address
ES: Ethernet Segment
ESI: Ethernet Segment Identifier
IRB: Integrated Routing and Bridging
LSP: Label Switched Path
MP2MP: Multipoint to Multipoint
MP2P: Multipoint to Point
ND: Neighbor Discovery
NA: Neighbor Advertisement
P2MP: Point to Multipoint
P2P: Point to Point
PE: Provider Edge
EVPN: Ethernet VPN
EVI: EVPN Instance
RT: Route Target

Single-Active Redundancy Mode: When only a single PE, among a group of PEs attached to an Ethernet segment, is allowed to forward traffic to/from that Ethernet Segment, then the Ethernet segment is defined to be operating in Single-Active redundancy mode.

All-Active Redundancy Mode: When all PEs attached to an Ethernet segment are allowed to forward traffic to/from that Ethernet Segment, then the Ethernet segment is defined to be operating in All-Active redundancy mode.

4. Routing Information Exchange for EVPN routes

[RFC7432] defines multiple Route Types NLRI along with procedures for

advertisements and processing of these routes. Some of these procedures are impacted as the result of hub-and-spoke architecture. The routing information exchange among the hub, spoke, and vanilla PEs are subject to the same rules as described in section 3 of [RFC7024]. Furthermore, if there are any changes to the EVPN route advisements and processing from that of [RFC7432], they are described below.

5. EVPN unknown MAC route

Section 3 of [RFC7024] talks about how a V-hub of a given VPN must export a VPN-IP default route for that VPN and this route must be exported to only the V-spokes of that VPN associated with that V-hub. [DCI-EVPN] defines the notion of the unknown MAC route for an EVI which is analogous to a VPN-IP default route for a VPN. This unknown MAC route is exported by a V-hub to its associated V-spokes. If multiple V-hubs are associated with a set of V-spokes, then each V-hub advertises it with a distinct RD when originating this route. If a V-spoke imports several of these unknown MAC routes and they all have the same preference, then traffic from the V-spoke to other sites of that EVI would be load balanced among the V-hubs.

5.1. Originating EVPN Unknown MAC Route by a V-Hub

Section 7.3 of the [RFC7024] defines procedures for originating a VPN-IP default route for a VPN. The same procuedures apply when a V-hub wants to originate EVPN unknown MAC route for a given EVI. The V-hub MUST announce unknown MAC route using the MAC/IP advertisement route along with the Default Gateway extended community as defined in section 10.1 of the [RFC7432].

5.2. Processing VPN-MAC EVPN unknown Route by a V-SPOKE

Within a given EVPN, a V-spoke MUST import all the unknown MAC routes unless the route-target mismatch happens. The processing of the received VPN-MAC EVPN default route follows the rules explained in the section 3 of the [RFC7024]. The unknown MAC route MUST be installed according to the rules of MAC/IP Advertisement route installation rules in section 9.2.2 of [RFC7024].

In absense of any more specific VPN-MAC EVPN routes, V-spokes installing the unknown MAC route MUST use the route when performing ARP proxy. This behavior would allow V-Spokes to forward the traffic towards V-Hub.

5.3. Aliasing

[RFC7432] describes the concept and procedures for Aliasing where a station is multi-homed to multiple PEs operating in an All-Active redundancy mode, it is possible that only a single PE learns a set of MAC addresses associated with traffic transmitted by the station. [RFC7432] describes the concepts and procedures for Aliasing, which occurs when a CE is multi-homed to multiple PE nodes, operating in all-active redundancy mode, but not all of the PEs learn the CE's set of MAC addresses. This leads to a situation where remote PEs receive MAC advertisement routes, for these addresses, from a single NVE even though multiple NVEs are connected to the multi-homed station. As a result, the remote NVEs are not able to effectively load-balance traffic among the NVEs connected to the multi-homed Ethernet segment.

To alleviate this issue, EVPN introduces the concept of Aliasing. This refers to the ability of a PE to signal that it has reachability to a given locally attached Ethernet segment, even when it has learnt no MAC addresses from that segment. The Ethernet A-D per-EVI route is used to that end. Remote PEs which receive MAC advertisement routes with non-zero ESI SHOULD consider the MAC address as reachable via all NVEs that advertise reachability to the relevant Segment using Ethernet A-D routes with the same ESI and with the Single-Active flag reset.

This procedure is impacted for virtual hub-and-spoke topology because a given V-spoke does not receive any MAC/IP advertisements from remote V-spokes; therefore, there is no point in propagating Ethernet A-D per-EVI route to the remote V-spokes. In this solution, the V-hubs terminate the Ethernet A-D per-EVI route (used for Aliasing) and follows the procedures described in [RFC7432] for handling this route.

There are scenarios for which it is desirable to establish direct communication path between a pair of V-spokes for a given host MAC address. In such scenario, the advertising V-spoke advertises both the MAC/IP route and Ethernet A-D per-EVI route with the RT of V-hub (RT-VH) per section 3 of [RFC7024]. The use of RT-VH, ensures that these routes are received by the V-spokes associated with that V-hub set and thus enables the V-spokes to perform the Aliasing procedure.

In summary, PE devices (V-hubs in general and V-spokes occasionally) that receive EVPN MAC/IP route advertisements (associated with a multi-homed site) need to also receive the associated Ethernet A-D per-EVI route advertisement(s) in order for them to perform Aliasing procedure.

5.4. Split-Horizon & Mass Withdraw

[RFC7432] uses Ethernet A-D per-ES route to a) signal to remote PEs the multi-homing redundancy type (Single-Active versus All-Active), b) advertise ESI label for split-horizon filtering when MPLS encapsulation is used, and c) advertise mass-withdraw when a failure of an access interface impacts many MAC addresses. This route does not need to be advertise from a V-spoke to any remote V-spoke unless a direct communication path between a pair of spoke is needed for a given flow.

Even if communication between a pair of V-spoke is needed for just a single flow, the Ethernet A-D per ES route needs to be advertised from the originating V-spoke for that ES which may handle tens or hundreds of thousands of flows. This is because in order to perform Aliasing function for a given flow, the Ethernet A-D per-EVI route is needed and this route itself is dependent on the Ethernet A-D per-ES route. In such scenario, the advertising V-spoke advertises the Ethernet A-D per-ES route with the RT of V-hub (RT-VH) per section 3 of [RFC7024].

In summary, PE devices (V-hubs in general and V-spokes occasionally) that receive EVPN MAC/IP route advertisements (associated with a multi-homed site) need to also receive the associated Ethernet A-D per-ES route advertisement(s).

6. Forwarding Considerations

6.1. IP-only Forwarding

When EVPN operates in IP-only forwarding mode using EVPN Route Type 5, then all forwarding considerations in section 4 of [RFC7024] are directly applicable here.

6.2. MAC-only Forwarding - Bridging

When EVPN operates in MAC-only forwarding mode (i.e., bridging mode), then for a given EVI, the MPLS label that a V-hub advertises with anUnknown MAC address MUST be the label that identifies the MAC-VRF of the V-hub in absense of a more specific MAC route. When the V-hub receives a packet with such label, the V- hub pops the label and determines further disposition of the packet based on the lookup in the MAC-VRF. Otherwise, the MPLS label of the matching more specific route is used and packet is is forwarded towards the associated NEXTHOP of the more specific route.

6.3. MAC and IP Forwarding - IRB

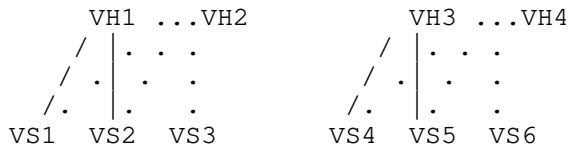
When a EVPN speaker operates in IRB mode, it implements both the IP and MAC forwarding Modes (aka Integrated Routing and Bridging - IRB). On a packet by packet basis, the V-spoke decides whether to do forwarding based on a MAC address lookup (bridge) or based on a IP address lookup (route). If the host destination MAC address is that of the IRB interface (i.e., if the traffic is inter-subnet), then the V-spoke performs an additional IP lookup in the IP-VRF. However, if the host destination MAC address is that of an actual host MAC address (i.e., the traffic is intra- subnet) , then the V-spoke only performs a MAC lookup in the MAC-VRF. The procedure specified in Section 6.1 and Section 6.2 are applicable to inter-subnet and intra-subnet forwarding respectively. For intra- subnet traffic, if the MAC address is not found in the MAC-VRF, then the V-spoke forwards the traffic to the V-hub with the MPLS label received from the V-hub for the unknown MAC address. For the Inter- subnet traffic, if the IP prefix is not found in the IP-VRF, then the V-spoke forwards the traffic to the V-hub with the MPLS label received from the V-hub for the default IP address.

7. Handling of Broadcast and Multicast traffic

Just like that V-spoke to V-spoke known unicast traffic is relayed by V-hubs, V-spoke to V-spoke BUM traffic can also relayed by V-hubs. This is especially desired if Ingress Replication (IR) would be used otherwise for V-spokes to send traffic to other V-spokes. This way, a V-spoke can unicast BUM traffic to a single V-hub, who will then relay the traffic. This achieves Assisted Replication, and reduces multicast state in the core. Note that a V-hub may relay traffic using MPLS P2MP tunnels or BIER as well as IR. While a V-spoke may use P2MP tunnels or BIER to send traffic to V-hubs, this specification focuses on using IR by V-spokes.

In this particular section, all traffic refers to BUM traffic unless explicitly stated otherwise. The term PE refers to a V-hub or V-spoke when there is no need to distinguish the two.

Consider the following topology, where V-spokes VS1/2/3 are associated with V-hubs VH1/2 in one cluster, and V-spokes VS4/5/6 are associated with V-hubs VH3/4 in another cluster. Note that the lines/dots in the diagram indcate association, not connection.



7.1. Split Horizon

When VH1 relays traffic that it receives from VS1, in case of IR it MUST not send traffic back to VS1, and in case of P2MP tunnel it must indicate that traffic is sourced from VS1 so that VS1 will discard the traffic. In case of IR with IP unicsat tunnels, the outer source IP address identifies the sending PE. In case of IR with MPLS unicast tunnels, VH1 must advertise different labels to different PEs, so that it can identify the sending PE based on the label in the traffic from a V-spoke.

If MPLS P2MP/multicast tunnels (including VXLAN-GPE and MPLS-over-GRE/UDP) are used by a V-hub to relay traffic, an upstream allocated (by the V-hub) label MUST be imposed in the label stack to identify the source of the V-spoke. The label is advertised as part of the PE Distinguisher (PED) Label Attribute of the Inclusive Multicast Ethernet Tag (IMET) route from the V-hub, as specified in Section 8 of [RFC 6514].

Notice that an "upstream-assigned" label used by a V-hub to send traffic with on a P2MP tunnel to identify the source V-spoke is the same "downstream-assigned" label used by the V-hub to receive traffic on the IR tunnel from the V-spoke. Therefore, the same PED Label attribute serves two purposes. With [RFC 6514], a PED label may only identify a PE but not a particular VPN. Here the PED label identifies both the PE and a particular EVI/BD. A V-spoke programs its context MPLS forwarding table for the V-hub to discard any traffic with the PED label that the V-hub advertised for this V-spoke, or pop other PED labels and direct traffic into a corresponding EVI for L2 forwarding.

Note that a V-hub cannot use VXLAN/NVGRE multicast tunnels to relay traffic because if the V-hub uses the source V-spoke's IP address in the outer IP header (for the purpose of identifying the source V-spoke), multicast RPF would fail and the packets will be discarded.

7.2. Route Advertisement

As with other route types, IMET routes from V-hubs are advertised with RT-VH and RT-EVI so they are imported by associated V-spokes and all V-hubs. They carry the PED Label attribute as described above.

IMET routes from V-spokes are advertised with RT-EVI so they are imported by all V-hubs. They also carry PED Label attribute for multi-homing split horizon purpose if and only if V-hubs uses IR to relay traffic.

If a V-hub uses RSVP-TE P2MP tunnel, IR, or BIER to send or relay traffic, all other PEs (V-hubs or V-spokes) will receive traffic directly because the V-hub sees all PEs. If a V-hub uses mLDP P2MP tunnel to send or relay traffic, only its associated V-spokes and all V-hubs will see the V-hub's IMET route and join the tunnel announced in the route. Another V-hub need to relay traffic to its associated V-spokes that are not associated with this V-hub.

For that V-hub to announce the mLDP relay tunnel in its cluster, it needs to advertise a (*,*) S-PMSI AD route, as specified in [BUM-PROCEDURE]. The route is advertised with the RT-VH for that cluster, and associated V-spokes will join the tunnel announced in the S-SPMI AD route.

7.3. Designated Forwarder in a Cluster

When there are multiple V-hubs in a cluster, a V-spoke in that cluster decides by itself to which V-hub to send traffic. If the receiving V-hub uses mLDP tunnel to relay traffic, V-hubs in other clusters need to further relay traffic, but only one V-hub in each cluster can do so. As a result, a DF must be elected among the V-hubs for each cluster.

The election is similar to DF election in RFC 7432, with the folllowing differences.

o Instead of using Ethernet Segment route to discover the PEs on a multi-homing ES, the IMET route are used to determine the V-hubs in the same cluster - they all carry the same pair of RT-EVI and RT-VH, and advertises the unknown mac route.

o Instead of using VLAN to do per-VLAN DF election, the Local Administration Field of the RT-EVI is used to do per-EVI DF election.

7.4. Traffic Forwarding Rules

When a PE needs to forward received traffic from local Attachment Circuits (ACs) or remote PEs to local ACs, it follows the rules in RFC 7432, except that traffic sourced from this local PE but relayed

back on a p2mp tunnel is discarded. It may also need to forward to other PEs, subject to rules in the following sections.

7.4.1. Traffic from Local ACs

Traffic from a V-hub's local ACs is forwarded using the tunnel announced in its IMET route, as specified in RFC 7432. In case of an mLDP tunnel, the traffic need to be relayed by V-hubs of other clusters to their associated V-spokes. For other tunnel types, no relay is needed.

Traffic from a V-spoke's local ACs is forwarded to an associated V-hub of its choice. In case of MPLS IR, the label in the V-hub's IMET route's PED attribute corresponding to this V-spoke is used.

7.4.2. Traffic Received by a V-hub from Another PE

When a V-hub receives traffic from an associated V-spoke, it needs to relay to other PEs, using the tunnel announced in its IMET route. In case of IR or BIER, the source V-spoke, which is determined from the incoming label or source IP address, is excluded from the replication list. In case of a P2MP tunnel, the popped incoming label is imposed again to identify the source PE, before the tunnel label is imposed.

When a V-hub receives traffic from another V-hub on a P2MP tunnel, and the tunnel is announced in an IMET route carrying the same RT-VH as this V-hub is configured with, it does not need to relay the traffic. Otherwise, the traffic is from a V-hub in a different cluster, and this V-hub needs to relay to its associated V-spokes, if and only if it is the DF for this cluster, using the tunnel announced in its (*,*) S-PMSI route carrying its RT-VH.

When a V-hub recevies traffic from another V-hub via IR or BIER, it does not further relay the traffic as that V-hub can reach all PEs.

7.4.3. Traffic received by a V-spoke from a V-hub

In case of P2MP tunnel, the V-spoke discards the traffic if the label following the tunnel label identifies the V-spoke itself.

7.5. Multi-homing support

Consider that an ES spans across two V-spokes in the same cluster and the V-hub uses MPLS IR to relay traffic. With ESI Label split horizon method, a source V-spoke uses the ESI label advertised by the V-hub for the ES, and the V-hub must change that to the ESI label advertised by receiving v-spokes when it relays traffic. That means V-hubs must advertise ESI labels for all multi-homing segments, even

when they're not on those segments. They must also do double label swap (EVI/BD label and ESI label) or mac lookup when relaying traffic.

There are two methods detailed below to avoid that complexity. Either one MAY be used.

7.5.1 Domain-wide Common Block (DCB) Label

[draft-zzhang-bess-mvpn-evpn-aggregation-label] proposes for all PEs on an MHES to use the same ESI label allocated from a Domain-wide Common Block. Not only does that have the advantages described in that document, but also It avoids the MHES complexity with Virtual Hub and Spoke as mentioned above, because the V-Hubs do not need to care about the ESI label at all any more.

7.5.2 Local Bias

If DCB labels cannot be used, then Local Bias can be used even For EVPN MPLS. The PED label following the mpls transport tunnel label or BIER header identifies the PE that originated the traffic in addition to identifying the EVI/BD.

If a V-hub uses P2MP or BIER to relay traffic, the PED label is one of the labels in the PE Distinguiser Label attribute in the V-hub's IMET route, allocated by the V-hub for the source V-spoke.

If a V-hub uses IR to relay traffic, for each V-spoke that it relays to, the PED label advertised by that receiving V-spoke for the source V-spoke needs to be imposed by the V-hub. For that purpose, each V-spoke must include the PED Label attribute in its IMET route, to advertise different labels for different PEs. It discovers the PEs that it needs to advertise labels for via the PED label Attribute in the V-hub's IMET route.

7.6. Direct V-spoke to V-spoke traffic

It may be desired for allow direct V-spoke to V-spoke traffic in a cluster, without the relay by a V-hub. To do that, V-spokes advertise their IMET routes with both RT-VH and RT-EVI. Forwarding rules will be specified in future revisions.

8. ARP/ND Suppression

[RFC7432] defines the procedures for ARP/ND suppression where a PE can terminate gratuitous ARP/ND request message from directly connected site and advertises the associated MAC and IP addresses in an EVPN MAC/IP advertisement route to all other remote PEs. The

remote PEs that receive this EVPN route advertisement, install the MAC/IP pair in their ARP/ND cache table thus enabling them to terminate ARP/ND requests and generate ARP/ND responses locally thus suppressing the flooding of ARP/ND requests over the EVPN network.

In this hub-and-spoke approach, the ARP suppression needs to be performed by both the EVPN V-hubs as well V-spokes as follow. When a V-Spoke receives a gratuitous ARP/ND request, it terminates it and stores the source MAC/IP pair in its ARP/ND cache table. Then, it advertises the source MAC/IP pair to its associated V-Hubs using EVPN MAC/IP advertisement route. The V-Hubs upon receiving this EVPN route advertisement, create an entry in their ARP/ND cache table for this MAC/IP pair.

Now when a V-Spoke receives an ARP/ND request, it first looks up its ARP cache table, if an entry for that MAC/IP pair is found, then an ARP/ND response is generated locally and sent to the CE. However, if an entry is not found, then the ARP/ND request is unicasted to one of the V-hub associated with this V-spoke. Since, the associated V-hub keeps all the MAC/IP ARP entries in its cache table, it can formulate and ARP/ND response and forward it to that CE via the corresponding V-spoke.

9. IANA Considerations

There is no additional IANA considerations for PBB-EVPN beyond what is already described in [RFC7432].

10. Security Considerations

All the security considerations in [RFC7432] apply directly to this document because this document leverages [RFC7432] control plane and their associated procedures - although not the complete set but rather a subset.

This draft does not introduce any new security considerations beyond that of [RFC7432] and [RFC4761] because advertisements and processing of B-MAC addresses follow that of [RFC7432], and processing of C-MAC addresses follow that of [RFC4761] - i.e, B-MAC addresses are learned in control plane and C-MAC addresses are learned in data plane.

11. Acknowledgements

The authors would like to thank Yakov Rekhter for initial idea discussions.

12. Change Log

Initial Version: Sep 21 2014 Original Name: draft-keyupate-evpn-virtual-hub-00.txt

13. References

13.1. Normative References

[RFC7024] Jeng, H., Uttaro, J., Jalil, L., Decraene, B., Rekhter, Y., and R. Aggarwal, "Virtual Hub-and-Spoke in BGP/MPLS VPNs", RFC 7024, October 2013.

[RFC7432] A. Sajassi, et al., "BGP MPLS Based Ethernet VPN", RFC 7432 , February 2015.

13.2. Informative References

[RFC7080] A. Sajassi, et al., "Virtual Private LAN Service (VPLS) Interoperability with Provider Backbone Bridges", RFC 7080, December 2013.

[RFC7209] D. Thaler, et al., "Requirements for Ethernet VPN (EVPN)", RFC 7209, May 2014.

[RFC4389] A. Sajassi, et al., "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, April 2006.

[RFC4761] K. Kompella, et al., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, Jauary 2007.

[OVERLAY] A. Sajassi, et al., "A Network Virtualization Overlay Solution using EVPN", draft-ietf-bess-evpn-overlay-01, work in progress, February 2015.

14. Authors' Addresses

Keyur Patel
Arrcus, Inc.
2077 Gateway Pl, Suite 400
San Jose, CA 95110, US
Email: keyur@arrcus.com

Ali Sajassi

Cisco
170 West Tasman Drive
San Jose, CA 95134, US
Email: sajassi@cisco.com

Yakov Rekhter
Juniper Networks, Inc.
Email: yakov@juniper.net

John E. Drake
Juniper Networks, Inc.
Email: jdrake@juniper.net

Zhaohui Zhang
Juniper Networks, Inc.
Email: zzhang@juniper.net

Wim Henderickx
Nokia
Email: wim.henderickx@nokia.com

BESS WorkGroup
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2019

Ali. Sajassi
Mankamana. Mishra
Samir. Thoria
Patrice. Brissette
Cisco Systems

October 22, 2018

AC-Aware Bundling Service Interface in EVPN
draft-sajassi-bess-evpn-ac-aware-bundling-00

Abstract

EVPN provides an extensible and flexible multi-homing VPN solution over an MPLS/IP network for intra-subnet connectivity among Tenant Systems and End Devices that can be physical or virtual.

EVPN multihoming with IRB is one of the common deployment scenarios. There are deployments which requires capability to have multiple subnets designated with multiple VLAN IDs in single bridge domain.

RFC7432 defines three different type of service interface which serve different requirements but none of them address the requirement to be able to support multiple subnets within single bridge domain. In this draft we define new service interface type to support multiple subnets in single bridge domain.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on April 25, 2019.

Table of Contents

1. Introduction

EVPN based multi-homing is becoming the basic building block for providing redundancy in next generation data center deployments as well as service provider access/aggregation network. For EVPN IRB mode, there are deployments which expect to be able to support multiple subnets within single Bridge Domain. Each subnets would be differentiated by VLAN. Thus, single IRB interface can still serve multiple subnets.

Motivation behind such deployments are

1. Manageability: If there is support to have multiple subnets using single bridge domain, it would require only one Bridge domain and one IRB for "N" subnets compare to "N" Bridge domain and "N" IRB interfaces to manage.

2. Simplicity: It avoids extra configuration by configuring Vlan Range as compare to individual VLAN, BD and IRB interface per subnet.

Multiple subnet per bridge domain deployments require that there would not be duplicate MAC address across subnet.

[RFC7432] defines three types of service interfaces. None of them provide flexibility to achieve multiple subnet within single bridge domain. Brief about existing service interface from [RFC7432] are ,

1. VLAN-Based Service Interface: With this service interface, an EVPN instance consists of only a single broadcast domain (e.g., a single VLAN). Therefore, there is a one-to-one mapping between a VID on this interface and a MAC-VRF.

2. VLAN Bundle Service Interface: With this service interface, an EVPN instance corresponds to multiple broadcast domains (e.g., multiple VLANs); however, only a single bridge table is maintained per MAC-VRF, which means multiple VLANs share the same bridge table. The MPLS-encapsulated frames MUST remain tagged with the originating VID. Tag translation is NOT permitted. The Ethernet Tag ID in all EVPN routes MUST be set to 0.

3. VLAN-Aware Bundle Service Interface: With this service interface, an EVPN instance consists of multiple broadcast domains (e.g., multiple VLANs) with each VLAN having its own bridge table -- i.e., multiple bridge tables (one per VLAN) are maintained by a single MAC-VRF corresponding to the EVPN instance.

Though from definition it looks like VLAN Bundle Service Interface does provide flexibility to support multiple subnet within single bridge domain. But it can not serve the requirement which is being described in this draft. For example, lets take the case from Figure-1, If PE1 learns MAC of H1 on Vlan 1 (subnet S1). When MAC route is originated , as per [RFC7432] ether tag would be set to 0. If there is packet coming from IRB interface which is untagged packet, and it reaches to PE2, PE2 does not have associated AC information. In this case PE2 can not forward traffic which is destined to H1.

This draft proposes an extension to existing service interface types defined in [RFC7432] and defines AC-aware Bundling service interface. AC-aware Bundling service interface would provide mechanism to have multiple subnets in single bridge domain. This extension is applicable only for multi-homed EVPN peers.

With this proposal IRB interface could either have multiple subnets or an aggregate subnet representing all individual subnets (when such aggregation is possible).

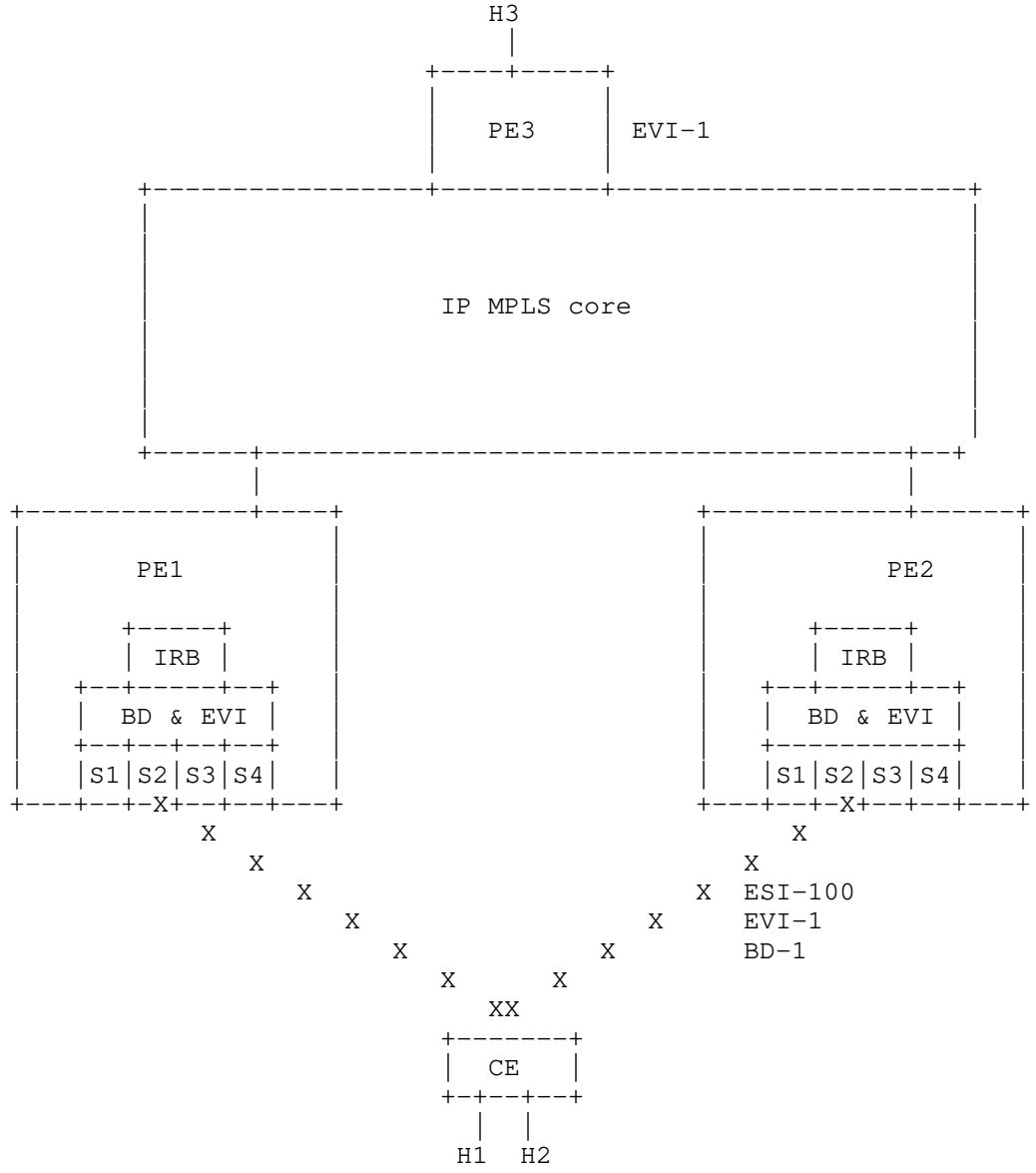


Figure 1: EVPN topology with multi-homing and non multihoming peer

The above figure shows sample EVPN topology, PE1 and PE2 are multihomed peers. PE3 is remote peer which is part of same EVPN instance (evi1). It is showing four subnets S1, S2, S3, S4 where numeric value provides associated Vlan information.

1.1. Problem with Unicast MAC route processing for multihome case

BD-1 has multiple subnets where each subnet is distinguished by Vlan 1, 2 ,3 and 4. PE1 learns MAC address MAC-1 from AC associated with subnet S1. PE1 uses MAC route to advertise MAC-1 presence to peer PEs. As per [RFC7432] MAC route advertisement from PE1 does not carry any context which can provide information about MAC address association with AC. When PE2 receives MAC route with MAC-2 it can not determine which AC this MAC belongs too.

Since PE2 could not bind MAC-1 with correct AC, when it receives data traffic destined to MAC-1, it can not find correct AC where data MUST be forwarded.

1.2. Problem with Multicast route synchronization

[I-D.ietf-bess-evpn-igmp-mld-proxy] defines mechanism to synchronize multicast routes between multihome peer. In above case if Receiver behind S1 send IGMP membership request, CE could hash it to either of the PE. When Multicast route is originated, it does not contain any AC information. Once it reaches to remote PE, it does not have any information about which subnet this IGMP membership request belong to.

1.3. Potential Security concern caused by misconfiguration

In case of single subnet per bridge domain, there is potential case of security issue. For example if PE1 , BD1 is configured with Vlan-1 where as multihome peer PE2 has configured Vlan-2. Now each of the IGMP membership request on PE1 would be synchronized to PE2. and PE2 would process multicast routes and start forwarding multicast traffic on Vlan-2, which was not intended.

2. Terminology

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

AC: Attachment Circuit.

ARP: Address Resolution Protocol.

BD: Broadcast Domain. As per [RFC7432], an EVI consists of a single or multiple BDs. In case of VLAN-bundle and VLAN-based service models (see [RFC7432]), a BD is equivalent to an EVI. In case of VLAN-aware bundle service model, an EVI contains multiple BDs. Also, in this document, BD and subnet are equivalent terms.

BD Route Target: refers to the Broadcast Domain assigned Route Target [RFC4364]. In case of VLAN-aware bundle service model, all the BD instances in the MAC-VRF share the same Route Target.

BT: Bridge Table. The instantiation of a BD in a MAC-VRF, as per [RFC7432].

DGW: Data Center Gateway.

Ethernet A-D route: Ethernet Auto-Discovery (A-D) route, as per [RFC7432].

Ethernet NVO tunnel: refers to Network Virtualization Overlay tunnels with Ethernet payload. Examples of this type of tunnels are VXLAN or GENEVE.

EVI: EVPN Instance spanning the NVE/PE devices that are participating on that EVPN, as per [RFC7432].

EVPN: Ethernet Virtual Private Networks, as per [RFC7432].

GRE: Generic Routing Encapsulation.

GW IP: Gateway IP Address.

IPL: IP Prefix Length.

IP NVO tunnel: it refers to Network Virtualization Overlay tunnels with IP payload (no MAC header in the payload)

IP-VRF: A VPN Routing and Forwarding table for IP routes on an NVE/PE. The IP routes could be populated by EVPN and IP-VPN address families. An IP-VRF is also an instantiation of a layer 3 VPN in an NVE/PE.

IRB: Integrated Routing and Bridging interface. It connects an IP-VRF to a BD (or subnet).

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on an NVE/PE, as per [RFC7432]. A MAC-VRF is also an instantiation of an EVI in an NVE/PE.

ML: MAC address length.

ND: Neighbor Discovery Protocol.

NVE: Network Virtualization Edge.

GENEVE: Generic Network Virtualization Encapsulation, [GENEVE].

NVO: Network Virtualization Overlays.

RT-2: EVPN route type 2, i.e., MAC/IP advertisement route, as defined in [RFC7432].

RT-5: EVPN route type 5, i.e., IP Prefix route. As defined in Section 3 of [EVPN-PREFIX].

SBD: Supplementary Broadcast Domain. A BD that does not have any ACs, only IRB interfaces, and it is used to provide connectivity among all the IP-VRFs of the tenant. The SBD is only required in IP-VRF- to-IP-VRF use-cases (see Section 4.4.).

SN: Subnet.

TS: Tenant System.

VA: Virtual Appliance.

VNI: Virtual Network Identifier. As in [RFC8365], the term is used as a representation of a 24-bit NVO instance identifier, with the understanding that VNI will refer to a VXLAN Network Identifier in VXLAN, or Virtual Network Identifier in GENEVE, etc. unless it is stated otherwise.

VTEP: VXLAN Termination End Point, as in [RFC7348].

VXLAN: Virtual Extensible LAN, as in [RFC7348].

This document also assumes familiarity with the terminology of [RFC7432],[RFC8365], [RFC7365].

3. Requirements

1. Service interface MUST be able to support multiple subnets designated by Vlan under single bridge domain.

2. New Service interface handling procedure MUST make sure to have backward compatibility with implementation procedures defined in [RFC7432]

3. New Service interface MUST be extendible to multicast routes defined in [I-D.ietf-bess-evpn-igmp-mld-proxy] too.

4. Solution Description

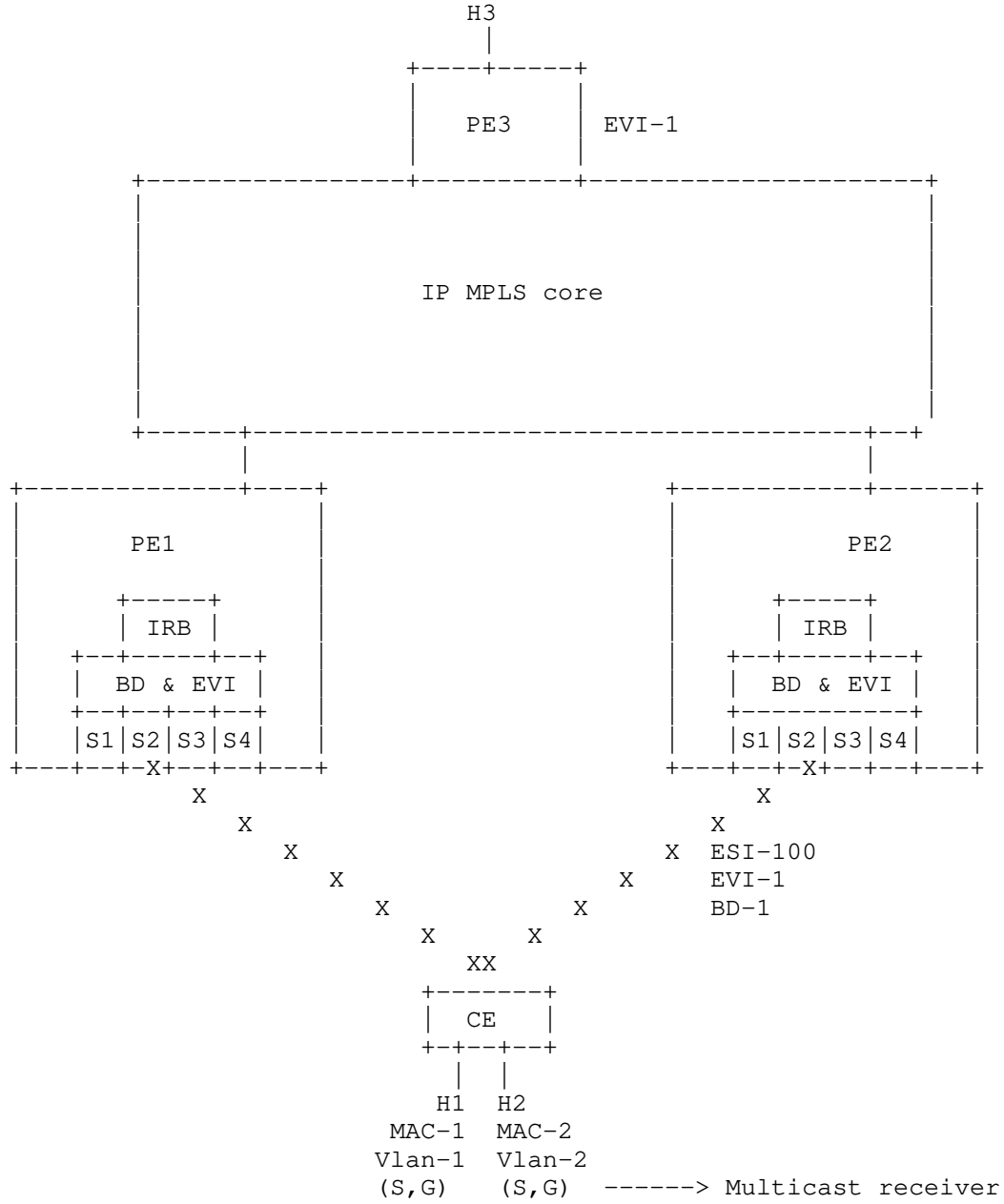


Figure 2: AC aware bundling procedures

Consider the above topology, where AC aware bundling service interface is supported. Host H1 on Vlan-1 has MAC address as MAC-1 and Host H2 on Vlan 2 has MAC address as MAC-2.

4.1. Control Plane Operation

4.1.1. MAC/IP Address Advertisement

4.1.1.1. Local Unicast MAC learning

1. [RFC7432] section 9.1 describes different mechanism to learn Unicast MAC address locally. PEs where AC aware bundling is supported, MAC address is learnt along with Vlan associated with AC.

2. MAC/IP route construction follows mechanism defined in [RFC7432] section 9.2.1. Along with RT-2 it must attach Attachment Circuit ID Extended Community (Section 5.1).

3. From Figure-2 PE1 learns MAC-1 on S1. It MUST construct MAC route with procedure defined in [RFC7432] section 9.2.1. It MUST attach Attachment Circuit ID Extended Community (Section 5.1).

4.1.1.2. Remote Unicast MAC learning

1. Presence of Attachment Circuit ID Extended Community (Section 5.1) MUST be ignored by non multihoming PEs. Remote PE (Non Multihome PE) MUST process MAC route as defined in [RFC7432]

2. Multihoming peer MUST process Attachment Circuit ID Extended Community (Section 5.1) to attach remote MAC address to appropriate AC.

3. From Figure-2 PE3 receives MAC route for MAC-1. It MUST ignore AC information in Attachment Circuit ID Extended Community (Section 5.1) which was received with RT-2.

4. PE2 receives MAC route for MAC-1. It MUST get Attachment Circuit ID from Attachment Circuit ID Extended Community (Section 5.1) in RT-2 and associate MAC address with specific subnet.

4.1.2. Multicast route Advertisement

4.1.2.1. Local multicast state

When a local multihomed bridge port in given BD receives IGMP membership request and ES is operating in All-active or Single-Active redundancy mode, it MUST synchronize multicast state by originating

multicast route defined in section 7 of [I-D.ietf-bess-evpn-igmp-mld-proxy]. When Service interface is AC aware it MUST attach Attachment Circuit ID Extended Community (Section 5.1) along with multicast route. For example in Figure-2 when H2 sends IGMP membership request for (S,G) , CE hashed it to one of the PE. Lets say PE1 received IGMP membership request, now PE1 MUST originate multicast route to synchronize multicast state with PE2. Multicast route MUST contain Attachment Circuit ID Extended Community (Section 5.1) along with multicast route.

If PE1 had already originated multicast route for (S,G) from subnet S2. Now if host H1 also sends IGMP membership request for (S,G) on subnet S1, PE1 MUST originate route update with Attachment Circuit ID Extended Community (Section 5.1).

4.1.2.2. Remote multicast state

If multihomed PE receives remote multicast route on Bridge Domain for given ES, route MUST be programmed to correct subnet. Subnet information MUST be get from Attachment Circuit ID Extended Community. For example PE2 receives multicast route on Bridge Domain BD-1 for ES ESI-100, From Attachment Circuit ID Extended Community (Section 5.1) it receives AC information and associates multicast route (S,G) to subnet S2.

When PE2 receives route update with Attachment Circuit ID Extended Community added for subnet S1, port associated with subnet S1 MUST be added for multicast route.

4.2. Data Plane Operation

4.2.1. Unicast Forwarding

1. Packet received from CE must follow same procedure as defined in [RFC7432] section 13.1

2. Unknown Unicast packets from a Remote PE MUST follow procedure as per [RFC7432] section 13.2.1.

3. Known unicast Received on a Remote PE MUST follow procedure as per [RFC7432] section 13.2.2. So in Figure-2 if PE3 receives known unicast packet for destination MAC MAC-1, it MUST follow procedure defined in [RFC7432] section 13.2.2.

4. If destination MAC lookup is performed on known unicast packet, destination MAC lookup MUST provide Vlan and Port tuple. For example if PE2 receives unicast packet which is destined to MAC-1 (packet might be coming from IRB or remote PE with EVPN tunnel),

destination MAC lookup on PE2 MUST provide outgoing port along with associated MAC address. In this case traffic MUST be forwarded to S1 with Vlan 1.

4.2.2. Multicast Forwarding

1. Multicast traffic from CE and remote PE MUST follow procedure defined in [RFC7432]

2. When multicast traffic is being received on IRB Interface, layer-3 forwarding is based on traditional multicast without any new modification. On bridge domain multicast traffic is forwarded towards right AC based on multicast state.

5. BGP Encoding

This document defines one new BGP Extended Community for EVPN.

5.1. Attachment Circuit ID Extended Community

A new EVPN BGP Extended Community called Attachment Circuit ID is introduced here. This new extended community is a transitive extended community with the Type field of 0x06 (EVPN) and the Sub-Type of TBD. It is advertised along with EVPN MAC/IP Advertisement Route (Route Type 2) per [RFC7432] for AC-Aware Bundling Service Interface.

The Attachment Circuit ID Extended Community is encoded as an 8-octet value as follows:

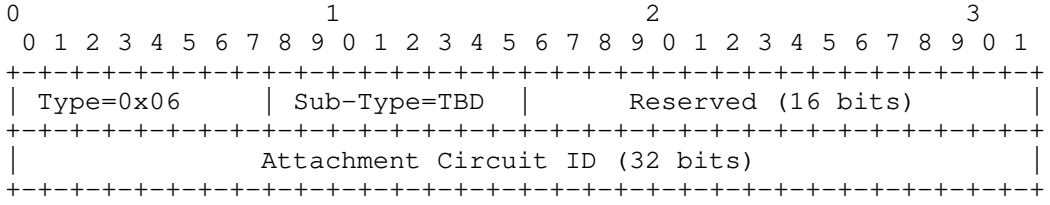


Figure 3: Attachment Circuit ID Extended Community

This extended community is used to carry the Attachment Circuit ID associated with the received MAC address and it is advertised along with EVPN MAC/IP and EVPN multicast Advertisement route. The receiving PE who is a member of an All-Active or Single-Active multi-homing group uses this information to not only synchronize the MAC address but also the associated AC over which the MAC addresses is received.

6. Security Considerations

The same Security Considerations described in [RFC7432] are valid for this document.

7. IANA Considerations

A new transitive extended community Type of 0x06 and Sub-Type of TBD for EVPN Attachment Circuit Extended Community needs to be allocated by IANA.

8. Acknowledgement

9. References

9.1. Normative References

[I-D.ietf-bess-evpn-igmp-mld-proxy]
Sajassi, A., Thoria, S., Patel, K., Yeung, D., Drake, J., and W. Lin, "IGMP and MLD Proxy for EVPN", draft-ietf-bess-evpn-igmp-mld-proxy-02 (work in progress), June 2018.

[I-D.ietf-bess-evpn-prefix-advertisement]
Rabadan, J., Henderickx, W., Drake, J., Lin, W., and A. Sajassi, "IP Prefix Advertisement in EVPN", draft-ietf-bess-evpn-prefix-advertisement-11 (work in progress), May 2018.

[I-D.ietf-idr-tunnel-encaps]
Rosen, E., Patel, K., and G. Velde, "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-10 (work in progress), August 2018.

9.2. Informative References

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

[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, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7365] Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for Data Center (DC) Network Virtualization", RFC 7365, DOI 10.17487/RFC7365, October 2014, <https://www.rfc-editor.org/info/rfc7365>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

Authors' Addresses

Ali Sajassi
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: sajassi@cisco.com

Mankamana Mishra
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: mankamis@cisco.com

Samir Thoria
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: sthoria@cisco.com

Patrice Brissette
Cisco Systems

Email: pbrisset@cisco.com

BESS WorkGroup
Internet-Draft
Intended status: Standards Track
Expires: February 19, 2021

A. Sajassi
M. Mishra
S. Thoria
P. Brissette
Cisco Systems
J. Rabadan
Nokia
J. Drake
Juniper Networks
August 18, 2020

AC-Aware Bundling Service Interface in EVPN
draft-sajassi-bess-evpn-ac-aware-bundling-02

Abstract

EVPN provides an extensible and flexible multi-homing VPN solution over an MPLS/IP network for intra-subnet connectivity among Tenant Systems and End Devices that can be physical or virtual.

EVPN multihoming with IRB is one of the common deployment scenarios. There are deployments which requires capability to have multiple subnets designated with multiple VLAN IDs in single bridge domain.

[RFC7432] defines three different type of service interface which serve different requirements but none of them address the requirement to be able to support multiple subnets within single bridge domain. In this draft we define new service interface type to support multiple subnets in single bridge domain. Service interface proposed in this draft will be applicable to multihoming case only.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on February 19, 2021.

Table of Contents

1. Introduction

EVPN based All-Active multi-homing is becoming the basic building block for providing redundancy in next generation data center deployments as well as service provider access/aggregation network. For EVPN IRB mode, there are deployments which expect to be able to support multiple subnets within single Bridge Domain. Each subnet would be differentiated by VLAN. Thus, single IRB interface can still serve multiple subnet.

Motivation behind such deployments are

1. Manageability: If there is support to have multiple subnets using single bridge domain, it would require only one Bridge domain and one IRB for "N" subnets compare to "N" Bridge domain and "N" IRB interface to manage.

2. Simplicity: It avoids extra configuration by configuring Vlan Range as compare to individual VLAN, BD and IRB interface per subnet.

Multiple subnet per bridge domain deployments guarantee that there would not be duplicate MAC address across subnet.

[RFC7432] defines three types of service interface. None of them provide flexibility to achieve multiple subnet within single bridge domain. Brief about existing service interface from [RFC7432] are ,

1. VLAN-Based Service Interface: With this service interface, an EVPN instance consists of only a single broadcast domain (e.g., a single VLAN). Therefore, there is a one-to-one mapping between a VID on this interface and a MAC-VRF.

2. VLAN Bundle Service Interface: With this service interface, an EVPN instance corresponds to multiple broadcast domains (e.g., multiple VLANs); however, only a single bridge table is maintained per MAC-VRF, which means multiple VLANs share the same bridge table. The MPLS-encapsulated frames MUST remain tagged with the originating VID. Tag translation is NOT permitted. The Ethernet Tag ID in all EVPN routes MUST be set to 0.

3. VLAN-Aware Bundle Service Interface: With this service interface, an EVPN instance consists of multiple broadcast domains (e.g., multiple VLANs) with each VLAN having its own bridge table -- i.e., multiple bridge tables (one per VLAN) are maintained by a single MAC-VRF corresponding to the EVPN instance.

Though from definition it looks like VLAN Bundle Service Interface does provide flexibility to support multiple subnet within single bridge domain. But it requirement is to have multiple subnet from same ES on multi-homing all active mode, it would not work. For example, lets take the case from Figure-1, If PE1 learns MAC of H1 on Vlan 1 (subnet S1). When MAC route is originated , as per [RFC7432] ether tag would be set to 0. If there is packet coming from IRB interface which is untagged packet, and it reaches to PE2, PE2 does not have associated AC information. In this case PE2 can not forward traffic which is destined to H1.

This draft proposes an extension to existing service interface types defined in [RFC7432] and defines AC-aware Bundling service interface. AC-aware Bundling service interface would provide mechanism to have multiple subnets in single bridge domain. This extension is applicable only for multi-homed EVPN peers..

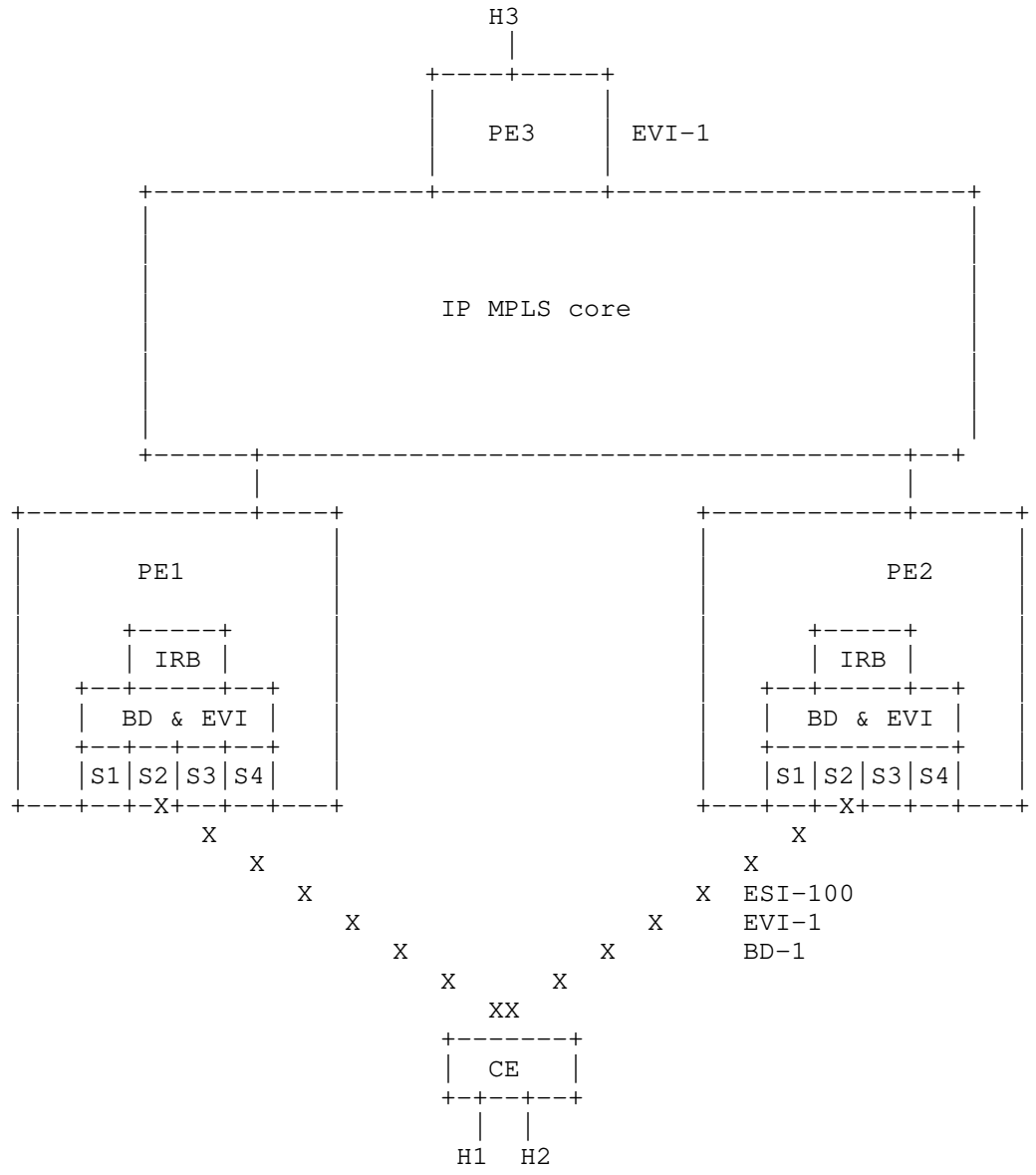


Figure 1: EVPN topology with multi-homing and non multihoming peer

The above figure shows sample EVPN topology, PE1 and PE2 are multihomed peers. PE3 is remote peer which is part of same EVPN instance (evi1). It is showing four subnets S1, S2, S3, S4 where numeric value provides associated Vlan information.

1.1. Problem with Unicast MAC route processing for multihome case

BD-1 has multiple subnet where each subnet is distinguished by Vlan 1, 2 ,3 and 4. PE1 learns MAC address MAC-1 from AC associated with subnet S1. PE1 uses MAC route to advertise MAC-1 presence to peer PEs. As per [RFC7432] MAC route advertisement from PE1 does not carry any context which can provide information about MAC address association with AC. When PE2 receives MAC route with MAC-2 it can not determine which AC this MAC belongs too.

Since PE2 could not bind MAC-1 with correct AC, when it receives data traffic destined to MAC-1, it can not find correct AC where data MUST be forwarded.

1.2. Problem with Multicast route synchronization

[I-D.ietf-bess-evpn-igmp-mld-proxy] defines mechanism to synchronize multicast routes between multihome peer. In above case if Receiver behind S1 send IGMP membership request, CE could hash it to either of the PE. When Multicast route is originated, it does not contain any AC information. Once it reaches to remote PE, it does not have any information about which subnet this IGMP membership request belong to.

1.3. Potential Security concern caused by misconfiguration

In case of single subnet per bridge domain, there is potential case of security issue. For example if PE1 , BD1 is configured with Vlan-1 where as multihome peer PE2 has configured Vlan-2. Now each of the IGMP membership request on PE1 would be synchronized to PE2. and PE2 would process multicast routes and start forwarding multicast traffic on Vlan-2, which was not intended.

2. Terminology

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

AC: Attachment Circuit.

ARP: Address Resolution Protocol.

BD: Broadcast Domain. As per [RFC7432], an EVI consists of a single or multiple BDs. In case of VLAN-bundle and VLAN-based service models (see [RFC7432]), a BD is equivalent to an EVI. In case of VLAN-aware bundle service model, an EVI contains multiple BDs. Also, in this document, BD and subnet are equivalent terms.

BD Route Target: refers to the Broadcast Domain assigned Route Target [RFC4364]. In case of VLAN-aware bundle service model, all the BD instances in the MAC-VRF share the same Route Target.

BT: Bridge Table. The instantiation of a BD in a MAC-VRF, as per [RFC7432].

DGW: Data Center Gateway.

Ethernet A-D route: Ethernet Auto-Discovery (A-D) route, as per [RFC7432].

Ethernet NVO tunnel: refers to Network Virtualization Overlay tunnels with Ethernet payload. Examples of this type of tunnels are VXLAN or GENEVE.

EVI: EVPN Instance spanning the NVE/PE devices that are participating on that EVPN, as per [RFC7432].

EVPN: Ethernet Virtual Private Networks, as per [RFC7432].

GRE: Generic Routing Encapsulation.

GW IP: Gateway IP Address.

IPL: IP Prefix Length.

IP NVO tunnel: it refers to Network Virtualization Overlay tunnels with IP payload (no MAC header in the payload)

IP-VRF: A VPN Routing and Forwarding table for IP routes on an NVE/PE. The IP routes could be populated by EVPN and IP-VPN address families. An IP-VRF is also an instantiation of a layer 3 VPN in an NVE/PE.

IRB: Integrated Routing and Bridging interface. It connects an IP-VRF to a BD (or subnet).

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on an NVE/PE, as per [RFC7432]. A MAC-VRF is also an instantiation of an EVI in an NVE/PE.

ML: MAC address length.

ND: Neighbor Discovery Protocol.

NVE: Network Virtualization Edge.

GENEVE: Generic Network Virtualization Encapsulation, [GENEVE].

NVO: Network Virtualization Overlays.

RT-2: EVPN route type 2, i.e., MAC/IP advertisement route, as defined in [RFC7432].

RT-5: EVPN route type 5, i.e., IP Prefix route. As defined in Section 3 of [EVPN-PREFIX].

SBD: Supplementary Broadcast Domain. A BD that does not have any ACs, only IRB interfaces, and it is used to provide connectivity among all the IP-VRFs of the tenant. The SBD is only required in IP-VRF- to-IP-VRF use-cases (see Section 4.4.).

SN: Subnet.

TS: Tenant System.

VA: Virtual Appliance.

VNI: Virtual Network Identifier. As in [RFC8365], the term is used as a representation of a 24-bit NVO instance identifier, with the understanding that VNI will refer to a VXLAN Network Identifier in VXLAN, or Virtual Network Identifier in GENEVE, etc. unless it is stated otherwise.

VTEP: VXLAN Termination End Point, as in [RFC7348].

VXLAN: Virtual Extensible LAN, as in [RFC7348].

This document also assumes familiarity with the terminology of [RFC7432],[RFC8365], [RFC7365].

3. Requirements

1. Service interface MUST be able to support multiple subnets designated by Vlan under single bridge domain.

2. Service interface MUST be applicable to Multihomed peers only

3. New Service interface handling procedure MUST make sure to have backward compatibility with implementation procedures defined in [RFC7432]

4. New Service interface MUST be extendible to multicast routes defined in [I-D.ietf-bess-evpn-igmp-mld-proxy] too.

4. Solution Description

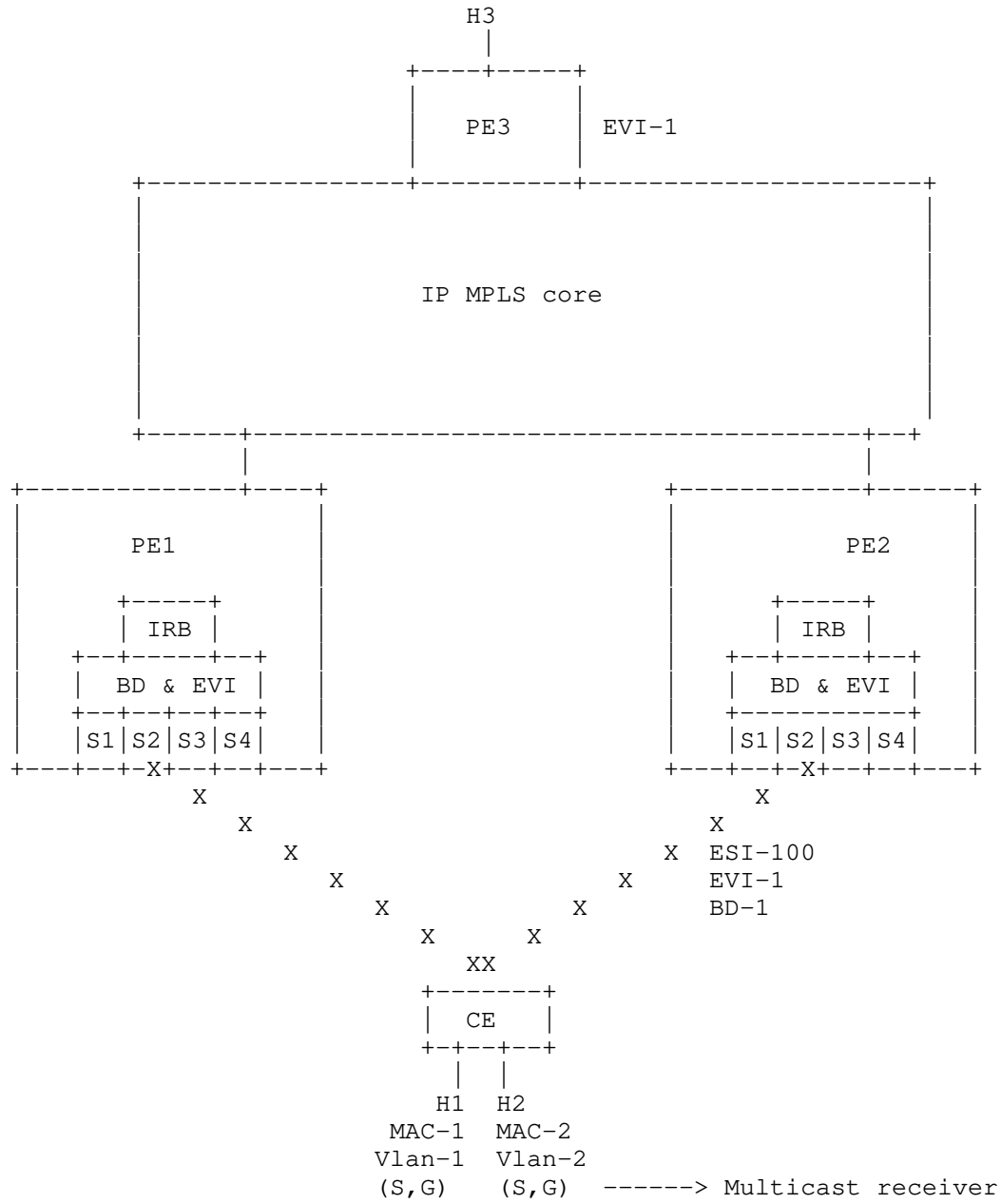


Figure 2: AC aware bundling procedures

Consider the above topology, where AC aware bundling service interface is supported. Host H1 on Vlan-1 has MAC address as MAC-1 and Host H2 on Vlan 2 has MAC address as MAC-2.

4.1. Control Plane Operation

4.1.1. MAC/IP Address Advertisement

4.1.1.1. Local Unicast MAC learning

1. [RFC7432] section 9.1 describes different mechanism to learn Unicast MAC address locally. PEs where AC aware bundling is supported, MAC address is learnt along with Vlan associated with AC.

2. MAC/IP route construction follows mechanism defined in [RFC7432] section 9.2.1. Along with RT-2 it must attach Attachment Circuit ID Extended Community (Section 6.1).

3. From Figure-2 PE1 learns MAC-1 on S1. It MUST construct MAC route with procedure defined in [RFC7432] section 9.2.1. It MUST attach Attachment Circuit ID Extended Community (Section 6.1).

4.1.1.2. Remote Unicast MAC learning

1. Presence of Attachment Circuit ID Extended Community (Section 6.1) MUST be ignored by non multihoming PEs. Remote PE (Non Multihome PE) MUST process MAC route as defined in [RFC7432]

2. Multihoming peer MUST process Attachment Circuit ID Extended Community (Section 6.1) to attach remote MAC address to appropriate AC.

3. From Figure-2 PE3 receives MAC route for MAC-1. It MUST not ignore AC information in Attachment Circuit ID Extended Community (Section 6.1) which was received with RT-2.

4. PE2 receives MAC route for MAC-1. It MUST get Attachment Circuit ID from Attachment Circuit ID Extended Community (Section 6.1) in RT-2 and associate MAC address with specific subnet.

4.1.2. Multicast route Advertisement

4.1.2.1. Local multicast state

When a local multihomed bridge port in given BD receives IGMP membership request and ES is operating in All-active or Single-Active redundancy mode, it MUST synchronize multicast state by originating

multicast route defined in section 7 of [I-D.ietf-bess-evpn-igmp-mld-proxy]. When Service interface is AC aware it MUST attach Attachment Circuit ID Extended Community (Section 6.1) along with multicast route. For example in Figure-2 when H2 sends IGMP membership request for (S,G) , CE hashed it to one of the PE. Lets say PE1 received IGMP membership request, now PE1 MUST originate multicast route to synchronize multicast state with PE2. Multicast route MUST contain Attachment Circuit ID Extended Community (Section 6.1) along with multicast route.

If PE1 had already originated multicast route for (S,G) from subnet S2. Now if host H1 also sends IGMP membership request for (S,G) on subnet S1, PE1 MUST originate route update with Attachment Circuit ID Extended Community (Section 6.1).

4.1.2.2. Remote multicast state

If multihomed PE receives remote multicast route on Bridge Domain for given ES, route MUST be programmed to correct subnet. Subnet information MUST be get from Attachment Circuit ID Extended Community. For example PE2 receives multicast route on Bridge Domain BD-1 for ES ESI-100, From Attachment Circuit ID Extended Community (Section 6.1) it receives AC information and associates multicast route (S,G) to subnet S2.

When PE2 receives route update with Attachment Circuit ID Extended Community added for subnet S1, port associated with subnet S1 MUST be added for multicast route.

4.2. Data Plane Operation

4.2.1. Unicast Forwarding

1. Packet received from CE must follow same procedure as defined in [RFC7432] section 13.1

2. Unknown Unicast packets from a Remote PE MUST follow procedure as per [RFC7432] section 13.2.1.

3. Known unicast Received on a Remote PE MUST follow procedure as per [RFC7432] section 13.2.2. So in Figure-2 if PE3 receives known unicast packet for destination MAC MAC-1, it MUST follow procedure defined in [RFC7432] section 13.2.2.

4. If destination MAC lookup is performed on known unicast packet, destination MAC lookup MUST provide Vlan and Port tuple. For example if PE2 receives unicast packet which is destined to MAC-1 (packet might be coming from IRB or remote PE with EVPN tunnel),

destination MAC lookup on PE2 MUST provide outgoing port along with associated MAC address. In this case traffic MUST be forwarded to S1 with Vlan 1.

4.2.2. Multicast Forwarding

1. Multicast traffic from CE and remote PE MUST follow procedure defined in [RFC7432]

2. Multicast traffic received from IRB interface or EVPN tunnel, route lookup would be performed based on IGMP snooping state and traffic would be forwarded to appropriate AC.

5. Mis-configuration of VLAN ranges across multihoming peer

If there is mis-configuration of Vlan or Vlan range across multihoming peer, same MAC address would be learnt with different Vlan in Bridge Domain. In this case Error message MUST be thrown for operator to make configuration changes. errored MAC route MUST be ignored.

6. BGP Encoding

This document defines one new BGP Extended Community for EVPN.

6.1. Attachment Circuit ID Extended Community

A new EVPN BGP Extended Community called Attachment Circuit ID is introduced here. This new extended community is a transitive extended community with the Type field of 0x06 (EVPN) and the Sub-Type of TBD. It is advertised along with EVPN MAC/IP Advertisement Route (Route Type 2) per [RFC7432] for AC-Aware Bundling Service Interface.

The Attachment Circuit ID Extended Community is encoded as an 8-octet value as follows:

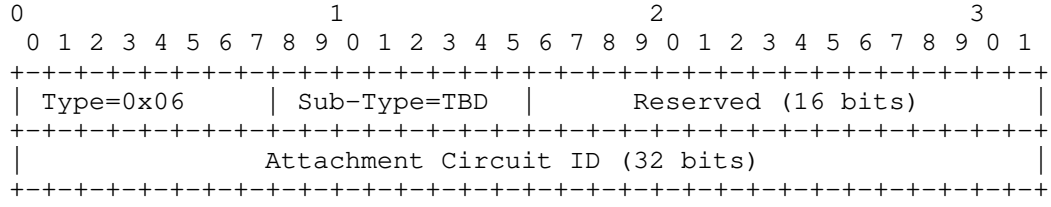


Figure 2: Attachment Circuit ID Extended Community

This extended community is used to carry the Attachment Circuit ID associated with the received MAC address and it is advertised along with EVPN MAC/IP Advertisement route. The receiving PE who is a member of an All-Active multi-homing group uses this information to not only synchronize the MAC address but also the associated AC over which the MAC addresses is received.

7. Security Considerations

The same Security Considerations described in [RFC7432] are valid for this document.

8. IANA Considerations

A new transitive extended community Type of 0x06 and Sub-Type of TBD for EVPN Attachment Circuit Extended Community needs to be allocated by IANA.

9. Acknowledgement

10. References

10.1. Normative References

[I-D.ietf-bess-evpn-igmp-mld-proxy]
Sajassi, A., Thoria, S., Patel, K., Yeung, D., Drake, J., and W. Lin, "IGMP and MLD Proxy for EVPN", draft-ietf-bess-evpn-igmp-mld-proxy-00 (work in progress), March 2017.

[I-D.ietf-bess-evpn-prefix-advertisement]
Rabadan, J., Henderickx, W., Drake, J., Lin, W., and A. Sajassi, "IP Prefix Advertisement in EVPN", draft-ietf-bess-evpn-prefix-advertisement-11 (work in progress), May 2018.

[I-D.ietf-idr-tunnel-encaps]
Rosen, E., Patel, K., and G. Velde, "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-10 (work in progress), August 2018.

10.2. Informative References

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

[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, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7365] Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for Data Center (DC) Network Virtualization", RFC 7365, DOI 10.17487/RFC7365, October 2014, <https://www.rfc-editor.org/info/rfc7365>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

Authors' Addresses

Ali Sajassi
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: sajassi@cisco.com

Mankamana Mishra
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: mankamis@cisco.com

Samir Thoria
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: sthoria@cisco.com

Patrice Brissette
Cisco Systems

Email: pbrisset@cisco.com

Jorge Rabadan
Nokia
777 E. Middlefield Road
Mountain View, CA 94043
UNITED STATES

Email: jorge.rabadan@nokia.com

John Drake
Juniper Networks

Email: jdrake@juniper.net

BESS Workgroup
INTERNET-DRAFT
Intended Status: Standards Track

A. Sajassi, Ed.
A. Banerjee
S. Thoria
D. Carrel
B. Weis
Cisco

Expires: May 20, 2019

October 20, 2018

Secure EVPN
draft-sajassi-bess-secure-evpn-00

Abstract

The applications of EVPN-based solutions ([RFC7432] and [RFC8365]) have become pervasive in Data Center, Service Provider, and Enterprise segments. It is being used for fabric overlays and inter-site connectivity in the Data Center market segment, for Layer-2, Layer-3, and IRB VPN services in the Service Provider market segment, and for fabric overlay and WAN connectivity in Enterprise networks. For Data Center and Enterprise applications, there is a need to provide inter-site and WAN connectivity over public Internet in a secured manner with same level of privacy, integrity, and authentication for tenant's traffic as IPsec tunneling using IKEv2. This document presents a solution where BGP point-to-multipoint signaling is leveraged for key and policy exchange among PE devices to create private pair-wise IPsec Security Associations without IKEv2 point-to-point signaling or any other direct peer-to-peer session establishment messages.

Table of Contents

Terminology

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

AC: Attachment Circuit.

ARP: Address Resolution Protocol.

BD: Broadcast Domain. As per [RFC7432], an EVI consists of a single or multiple BDs. In case of VLAN-bundle and VLAN-based service models (see [RFC7432]), a BD is equivalent to an EVI. In case of VLAN-aware bundle service model, an EVI contains multiple BDs. Also, in this document, BD and subnet are equivalent terms.

BD Route Target: refers to the Broadcast Domain assigned Route Target [RFC4364]. In case of VLAN-aware bundle service model, all the BD instances in the MAC-VRF share the same Route Target.

BT: Bridge Table. The instantiation of a BD in a MAC-VRF, as per [RFC7432].

DGW: Data Center Gateway.

Ethernet A-D route: Ethernet Auto-Discovery (A-D) route, as per [RFC7432].

Ethernet NVO tunnel: refers to Network Virtualization Overlay tunnels with Ethernet payload. Examples of this type of tunnels are VXLAN or GENEVE.

EVI: EVPN Instance spanning the NVE/PE devices that are participating on that EVPN, as per [RFC7432].

EVPN: Ethernet Virtual Private Networks, as per [RFC7432].

GRE: Generic Routing Encapsulation.

GW IP: Gateway IP Address.

IPL: IP Prefix Length.

IP NVO tunnel: it refers to Network Virtualization Overlay tunnels with IP payload (no MAC header in the payload).

IP-VRF: A VPN Routing and Forwarding table for IP routes on an NVE/PE. The IP routes could be populated by EVPN and IP-VPN address families. An IP-VRF is also an instantiation of a layer 3 VPN in an NVE/PE.

IRB: Integrated Routing and Bridging interface. It connects an IP-VRF to a BD (or subnet).

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on an NVE/PE, as per [RFC7432]. A MAC-VRF is also an instantiation of an EVI in an NVE/PE.

ML: MAC address length.

ND: Neighbor Discovery Protocol.

NVE: Network Virtualization Edge.

GENEVE: Generic Network Virtualization Encapsulation, [GENEVE].

NVO: Network Virtualization Overlays.

RT-2: EVPN route type 2, i.e., MAC/IP advertisement route, as defined in [RFC7432].

RT-5: EVPN route type 5, i.e., IP Prefix route. As defined in Section 3 of [EVPN-PREFIX].

SBD: Supplementary Broadcast Domain. A BD that does not have any ACs, only IRB interfaces, and it is used to provide connectivity among all the IP-VRFs of the tenant. The SBD is only required in IP-VRF- to-IP-VRF use-cases (see Section 4.4.).

SN: Subnet.

TS: Tenant System.

VA: Virtual Appliance.

VNI: Virtual Network Identifier. As in [RFC8365], the term is used as a representation of a 24-bit NVO instance identifier, with the understanding that VNI will refer to a VXLAN Network Identifier in VXLAN, or Virtual Network Identifier in GENEVE, etc. unless it is stated otherwise.

VTEP: VXLAN Termination End Point, as in [RFC7348].

VXLAN: Virtual Extensible LAN, as in [RFC7348].

This document also assumes familiarity with the terminology of [RFC7432], [RFC8365] and [RFC7365].

1 Introduction

The applications of EVPN-based solutions have become pervasive in Data Center, Service Provider, and Enterprise segments. It is being used for fabric overlays and inter-site connectivity in the Data Center market segment, for Layer-2, Layer-3, and IRB VPN services in the Service Provider market segment, and for fabric overlay and WAN connectivity in the Enterprise networks. For Data Center and Enterprise applications, there is a need to provide inter-site and WAN connectivity over public Internet in a secured manner with the same level of privacy, integrity, and authentication for tenant's traffic as used in IPsec tunneling using IKEv2. This document presents a solution where BGP point-to-multipoint signaling is leveraged for key and policy exchange among PE devices to create private pair-wise IPsec Security Associations without IKEv2 point-to-point signaling or any other direct peer-to-peer session establishment messages.

EVPN uses BGP as control-plane protocol for distribution of information needed for discovery of PEs participating in a VPN, discovery of PEs participating in a redundancy group, customer MAC addresses and IP prefixes/addresses, aliasing information, tunnel encapsulation types, multicast tunnel types, multicast group memberships, and other info. The advantages of using BGP control plane in EVPN are well understood including the following:

1) A full mesh of BGP sessions among PE devices can be avoided by using Route Reflector (RR) where a PE only needs to setup a single BGP session between itself and the RR as opposed to setting up N BGP sessions to N other remote PEs; therefore, reducing number of BGP sessions from O(N^2) to O(N) in the network. Furthermore, RR hierarchy can be leveraged to scale the number of BGP routes on the RR.

2) MP-BGP route filtering and constrained route distribution can be leveraged to ensure that the control-plane traffic for a given VPN is only distributed to the PEs participating in that VPN.

For setting up point-to-point security association (i.e., IPsec tunnel) between a pair of EVPN PEs, it is important to leverage BGP point-to-multipoint singling architecture using the RR along with its route filtering and constrain mechanisms to achieve the performance and the scale needed for large number of security associations (IPsec tunnels) along with their frequent re-keying requirements. Using BGP signaling along with the RR (instead of peer-to-peer protocol such as IKEv2) reduces number of message exchanges needed for SAs establishment and maintenance from O(N^2) to O(N) in the network. be increased from O(N) to O(N^2).

2 Requirements

The requirements for secured EVPN are captured in the following subsections.

2.1 Tenant's Layer-2 and Layer-3 data & control traffic

Tenant's layer-2 and layer-3 data and control traffic SHALL be protected by IPsec cryptographic methods. This implies not only tenant's data traffic SHALL be protected by IPsec but also tenant's control and routing information that are advertised in BGP SHALL also be protected by IPsec. This in turn implies that BGP session SHALL be protected by IPsec.

2.2 Tenant's Unicast & Multicast Data Protection

Tenant's layer-2 and layer-3 unicast traffic SHALL be protected by IPsec. In addition to that, tenant's layer-2 broadcast, unknown unicast, and multicast traffic as well as tenant's layer-3 multicast traffic SHALL be protected by IPsec when ingress replication or assisted replication are used. The use of BGP P2MP signaling for setting up P2MP SAs in P2MP multicast tunnels is for future study.

2.3 P2MP Signaling for SA setup and Maintenance

BGP P2MP signaling SHALL be used for IPsec SAs setup and maintenance. The BGP signaling SHALL follow P2MP signaling framework per [CONTROLLER-IKE] for IPsec SAs setup and maintenance in order to reduce the number of message exchanges from O(N^2) to O(N) among the participant PE devices.

2.3 Granularity of Security Association Tunnels

The solution SHALL support the setup and maintenance of IPsec SAs at the following level of granularities:

1) Per pair of PEs: A single IPsec tunnel between a pair of PEs to be used for all tenants' traffic supported by the pair of PEs.

2) Per tenant: A single IPsec tunnel per tenant per pair of PEs. For example, if there are 1000 tenants supported on a pair of PEs, then 1000 IPsec tunnels are required between that pair of PEs.

3) Per subnet: A single IPsec tunnel per subnet (e.g., per VLAN/EVI) of a tenant on a pair of PEs.

4) Per pair of IP addresses: A single IPsec tunnel per pair of IP addresses of a tenant on a pair of PEs.

5) Per pair of MAC addresses: A single IPsec tunnel per pair of MAC addresses of a tenant on a pair of PEs.

2.4 Support for Policy and DH-Group List

The solution SHALL support a single policy and DH group for all SAs as well as supporting multiple policies and DH groups among the SAs.

3 Solution Description

This solution uses BGP P2MP signaling where an originating PE only send a message to Route Reflector (RR) and then the RR reflects that message to the interested recipient PEs. The framework for such signaling is described in [CONTROLLER-IKE] and it is referred to as device-to-controller trust model. This trust model is significantly different than the traditional peer-to-peer trust model where a P2P signaling protocol such as IKEv2 [RFC7296] is used in which the PE devices directly authenticate each other and agree upon security policy and keying material to protect communications between themselves. The device-to-controller trust model leverages P2MP signaling via the controller (e.g., the RR) to achieve much better scale and performance for establishment and maintenance of large number of pairwise Security Associations (SAs) among the PEs.

This device-to-controller trust model first secures the control channel between each device and the controller using peer-to-peer protocol such as IKEv2 [RFC7296] to establish P2P SAs between each PE and the RR. It then uses this secured control channel for P2MP signaling in establishment of P2P SAs between a pair of PE devices.

Each PE advertised to other PEs via the RR the information needed in establishment of pair-wise SAs between itself an every other remote PEs. These pieces of information are sent as Sub-TLVs of IPSec tunnel type in BGP Tunnel Encapsulation attribute. These Sub-TLVs are detailed in section 5 and they are based on IKEv2 specification [RFC7296]. The IPsec tunnel TLVs along with its Sub-TLVs are sent along with the BGP route (NLRI) for a given level of granularity.

If only a single SA is required per pair of PE devices to multiplex user traffic for all tenants, then IPsec tunnel TLV is advertised along with IPv4 or IPv6 NLRI representing loopback address of the originating PE. It should be noted that this is not a VPN route but rather an IPv4 or IPv6 route.

If a SA is required per tenant between a pair of PE devices, then IPsec tunnel TLV can be advertised along with EVPN IMET route

representing the tenant or can be advertised along with a new EVPN route representing the tenant.

If a SA is required per tenant's subnet (e.g., per VLAN) between a pair of PE devices, then IPsec tunnel TLV is advertised along with EVPN IMET route.

If a SA is required between a pair of tenant's devices represented by a pair of IP addresses, then IPsec tunnel TLV is advertised along with EVPN IP Prefix Advertisement Route or EVPN MAC/IP Advertisement route.

If a SA is required between a pair of tenant's devices represented by a pair of MAC addresses, then IPsec tunnel TLV is advertised along with EVPN MAC/IP Advertisement route.

If a SA is required between a pair of tenant's devices represented by a VLAN or a port, then IPsec tunnel TLV is advertised along with EVPN Ethernet AD route.

3.1 Distribution of Public Keys and Policies

One of the requirements for this solution is to support a single DH group and a single policy for all SAs as well as to support multiple DH groups and policies among the SAs. The following subsections describe what pieces of information (what Sub-TLVs) are needed to be exchanged to support a single DH group and a single policy versus multiple DH groups and multiple policies.

3.1.1 Minimum Set

For SA establishment, at the minimum, a PE needs to advertise to other PEs, its ID, a notification to indicate if this is its initial contact, key exchange including DH public number and DH group, and Nonce. When a single policy is used among all SAs, it is assumed that this single policy is configured by the management system in all the PE devices and thus there is no need to signal it. The information that need to be signaled (using RFC7296 notations) are:

ID, [N(INITIAL_CONTACT),] KE, Ni; where

ID payload is defined in section 3.5 of [RFC7296]
N (Notify) Payload in section 3.10 of [RFC7296]
KE (Key Exchange) payload in section 3.4 of [RFC7296]
Ni (Nonce) payload in section 3.9 of [RFC7296]

KE payload contains the DH public number and also identifies which DH

group to use. ID sub-TLV would not be needed in BGP because tunnel attribute already carries originator ID. Section 5 details these sub-TLVs as part of IPsec tunnel TLV in BGP Tunnel Encapsulation Attribute.

3.1.2 Single Policy

If a single policy needs to be signaled among per tenant or per subnet among a set of PEs, then in addition to the information described in section 3.1.1, Security Association sub-TLV needs to be signaled as well. The payload for this sub-TLV is defined in section 3.3 of [RFC7296] and detailed in section 5.3.

ID, [N(INITIAL_CONTACT),SA, KE, Ni

SA (Security Association) payload in section 3.3 of [RFC7296]

A single SA payload identifies a single IPsec policy. One important restriction on the SA Payload is that an standard IKE SA payload can contain multiple transform; however, [CONTROLLER-IKE] restricts the SA payload to only a single transform for each transform type as described in section A.3.1 of [CONTROLLER-IKE].

3.1.3 Policy-list & DH-group-list

There can be scenarios for which there is a need to have multiple policy options. This can happen when there is a need for policy change and smooth migration among all PE devices to the new policy is required. It can also happen if different PE devices have different capabilities within the network. In these scenairos, PE devices need to be able to choose the correct policy to use for each other. This multi-policy scheme is described in section 6 of [CONTROLLER-IKE]. In order to support this multi-policy feature, a PE device MUST distribute a policy list. This list consists of multiple distinct policies in order of preference, where the first policy is the most preferred one. The receiving PE selects the policy by taking the received list (starting with the first policy) and comparing that against its own list and choosing the first one found in common. If there is no match, this indicates a configuration error and the PEs MUST NOT establish new SAs until a message is received that does produce a match.

Furthermore, when a device supports more than one DH group, then a unique DH public number MUST be specified for each in order of preference. The selection of which DH group to use follows the same logic as Policy selection, using the receiver’s list order until a

match is found in the initiator's list.

In order to support multi-policy a policy list is signaled in addition to the information described in section 3.1.1. Furthermore, in order to support multi-DH-groups, a DH group list along with its nonce list are signaled instead of a single DH group and a single nonce as described in section 3.1.1.

ID, [N(INITIAL_CONTACT), [SA], [KE], [Ni]

[SA] list of IPsec policies (i.e., list of SA payloads)
[KE] list of KE payloads

3.2 Initial IPsec SAs Generation

The procedure for generation of initial IPsec SAs is described in section 3 of [CONTROLLER-IKE]. This section gives a summary of it in context of BGP signaling. When a PE device first comes up and wants to setup an IPsec SA between itself and each of the interested remote PEs, it generates a DH pair along for each of its intended IPsec SA using an algorithm defined in the IKEv2 Diffie-Hellman Group Transform IDs [IKEv2-IANA]. The originating PE distributes DH public value along with a nonce (using IPsec Tunnel TLV in Tunnel Encapsulation Attribute) to other remote PEs via the RR. Each receiving PE uses this DH public number and the corresponding nonce in creation of IPsec SA pair to the originating PE – i.e., an outbound SA and an inbound SA. The detail procedures are described in section 5.2 of [CONTROLLER-IKE].

3.3 Re-Keying

A PE can initiate re-keying at any time due to local time or volume based policy or due to the result of cipher counter nearing its final value. The rekey process is performed individually for each remote PE. If rekeying is performed with multiple PEs simultaneously, then the decision process and rules described in this rekey are performed independently for each PE. Section 4 of [CONTROLLER-IKE] describes this rekeying process in details and gives examples for a single IPsec device (e.g., a single PE) rekey versus multiple PE devices rekey simultaneously.

3.4 IPsec Databases

The Peer Authorization Database (PAD), the Security Policy Database (SPD), and the Security Association Database (SAD) all need to be

setup as defined in the IPsec Security Architecture [RFC4301]. Section 5 of [CONTROLLER-IKE] gives a summary description of how these databases are setup for the controller-based model where key is exchanged via P2MP signaling via the controller (e.g., the RR) and the policy can be either signaled via the RR (in case of multiple policies) or configured by the management station (in case of single policy).

4 Encapsulation

Vast majority of Encapsulation for Network Virtualization Overlay (NVO) networks in deployment are based on UDP/IP with UDP destination port ID indicating the type of NVO encapsulation (e.g., VxLAN, GPE, GENEVE, GUE) and UDP source port ID representing flow entropy for load-balancing of the traffic within the fabric based on n-tuple that includes UDP header. When encrypting NVO encapsulated packets using IP Encapsulating Security Payload (ESP), the following two options can be used: a) adding a UDP header before ESP header (e.g., UDP header in clear) and b) no UDP header before ESP header (e.g., standard ESP encapsulation). The following subsection describe these encapsulation in further details.

4.1 Standard ESP Encapsulation

When standard IP Encapsulating Security Payload (ESP) is used (without outer UDP header) for encryption of NVO packets, it is used in transport mode as depicted below. When such encapsulation is used, the Tunnel Type of Tunnel Encapsulation TLV is set to ESP-Transport and the Tunnel Type of Encapsulation Extended Community is set to NVO encapsulation type (e.g., VxLAN, GENEVE, GPE, etc.). This implies that the customer packets are first encapsulated using NVO encapsulation type and then it is further encapsulated & encrypted using ESP-Transport mode.

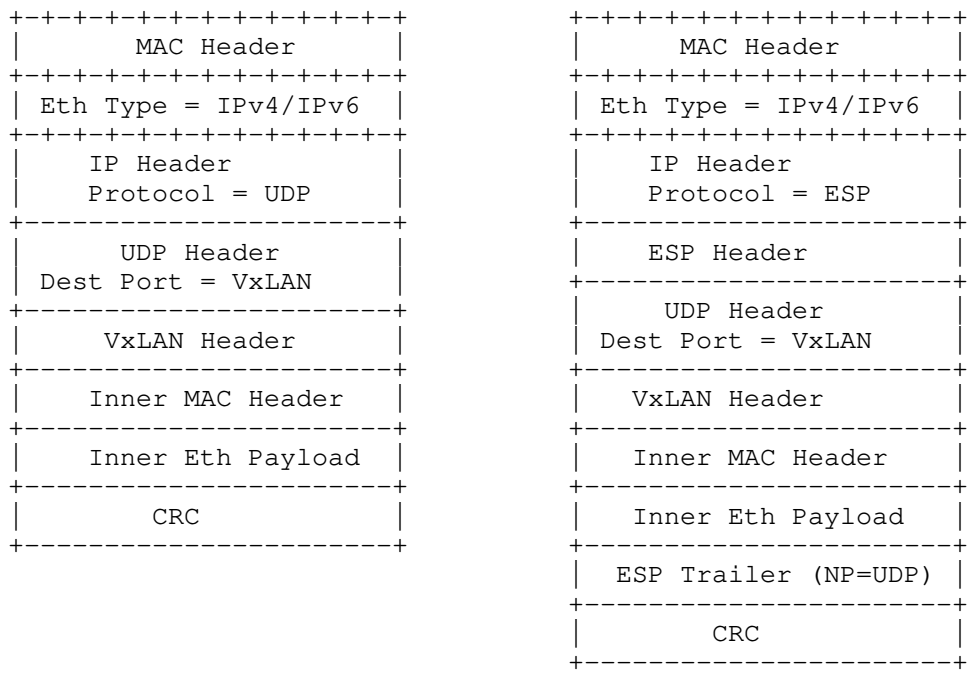


Figure 3: VxLAN Encapsulation within ESP

4.2 ESP Encapsulation within UDP packet

In scenarios where NAT traversal is required ([RFC3948]) or where load balancing using UDP header is required, then ESP encapsulation within UDP packet as depicted in the following figure is used. The ESP for NVO applications is in transport mode. The outer UDP header (before the ESP header) has its source port set to flow entropy and its destination port set to 4500 (indicating ESP header follows). A non-zero SPI value in ESP header implies that this is a data packet (i.e., it is not an IKE packet). The Next Protocol field in the ESP trailer indicates what follows the ESP header, is a UDP header. This inner UDP header has a destination port ID that identifies NVO encapsulation type (e.g., VxLAN). Optimization of this packet format where only a single UDP header is used (only the outer UDP header) is for future study.

When such encapsulation is used, the Tunnel Type of Tunnel Encapsulation TLV is set to ESP-in-UDP-Transport and the Tunnel Type of Encapsulation Extended Community is set to NVO encapsulation type

(e.g., VxLAN, GENEVE, GPE, etc.). This implies that the customer packets are first encapsulated using NVO encapsulation type and then it is further encapsulated & encrypted using ESP-in-UDP with Transport mode.

[RFC3948]

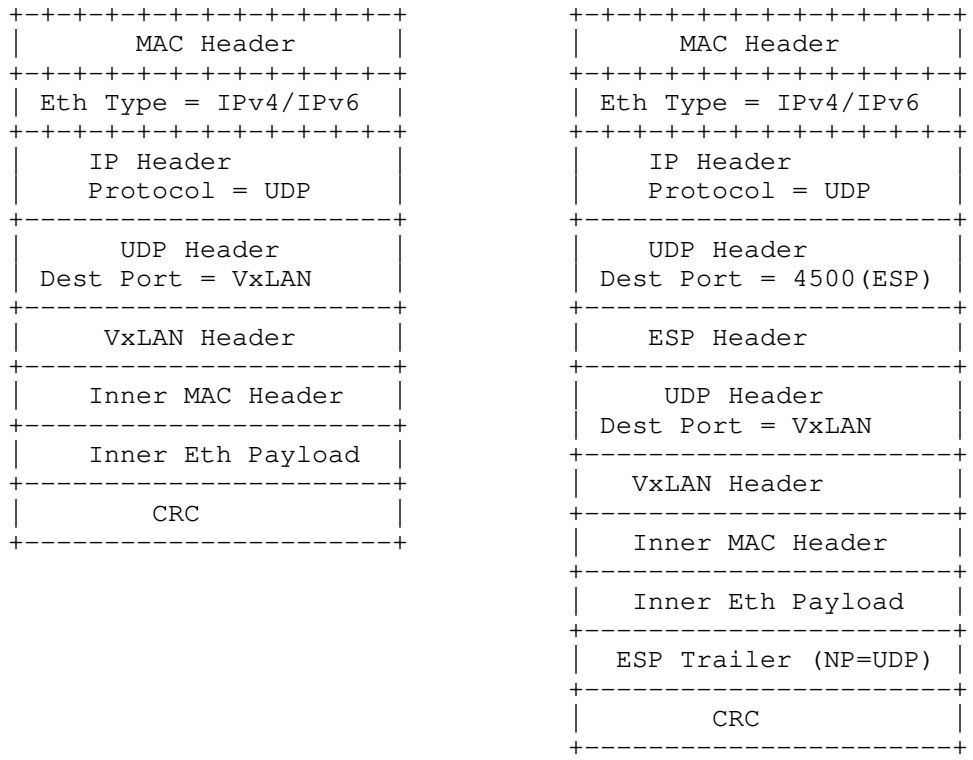


Figure 4: VxLAN Encapsulation within ESP Within UDP

5 BGP Encoding

This document defines two new Tunnel Types along with its associated sub-TLVs for The Tunnel Encapsulation Attribute [TUNNEL-ENCAP]. These tunnel types correspond to ESP-Transport and ESP-in-UDP-Transport as described in section 4. The following sub-TLVs apply to both tunnel types unless stated otherwise.

5.1 ESP Notify Sub-TLV

This sub-TLV corresponds to Notify payload of IPsec Encapsulation Security Payload protocol as defined in IKEv2 [RFC7296]. This payload is defined and described in section 3.10 of [RFC7296].

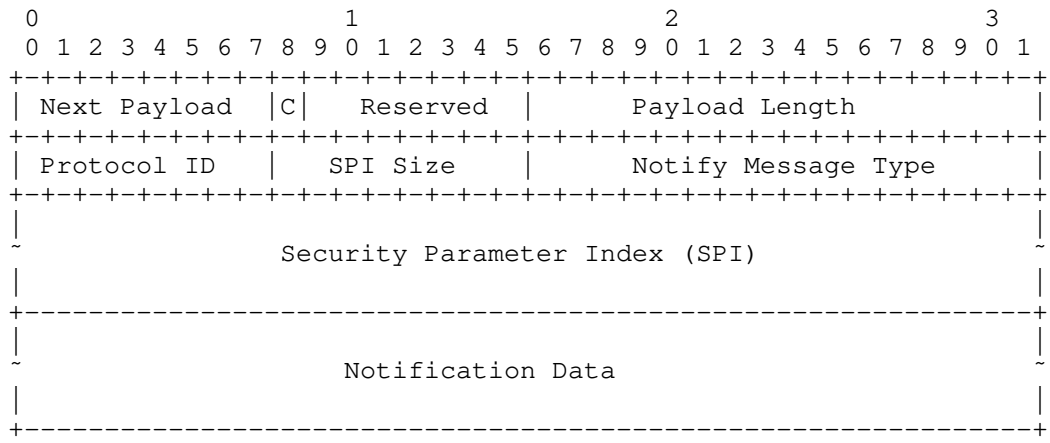


Figure 5: Notify Payload Format

5.2 ESP Key Exchange Sub-TLV

This sub-TLV corresponds to Key Exchange payload of IPsec Encapsulation Security Payload protocol as defined in IKEv2 [RFC7296]. This payload is defined and described in section 3.4 of [RFC7296].

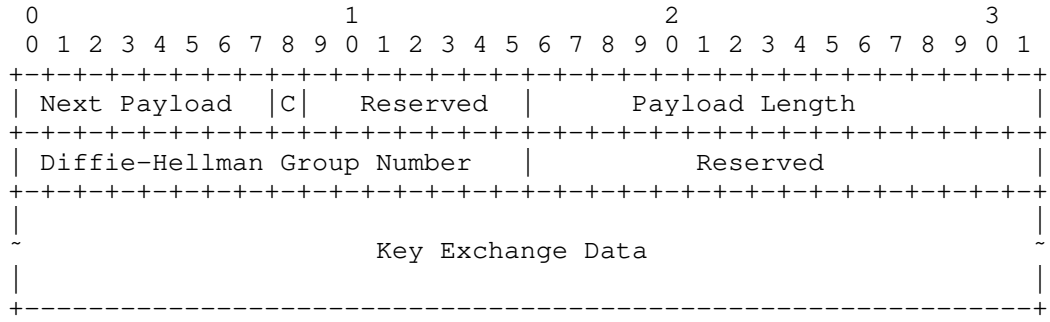


Figure 6: Key Exchange Payload Format

5.3 ESP Nonce Sub-TLV

This sub-TLV corresponds to Nonce payload of IPsec Encapsulation

Security Payload protocol as defined in IKEv2 [RFC7296]. This payload is defined and described in section 3.9 of [RFC7296].

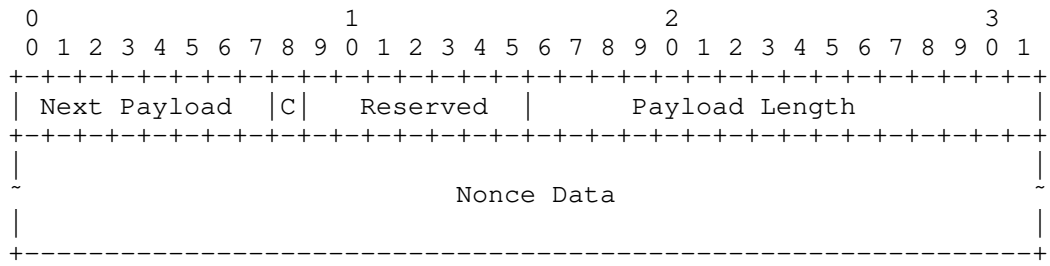


Figure 7: Nonce Payload Format

5.3 ESP Proposals Sub-TLV

This sub-TLV corresponds to Proposal payload of IPsec Encapsulation Security Payload protocol as defined in IKEv2 [RFC7296]. This payload is defined and described in section 3.3 of [RFC7296].

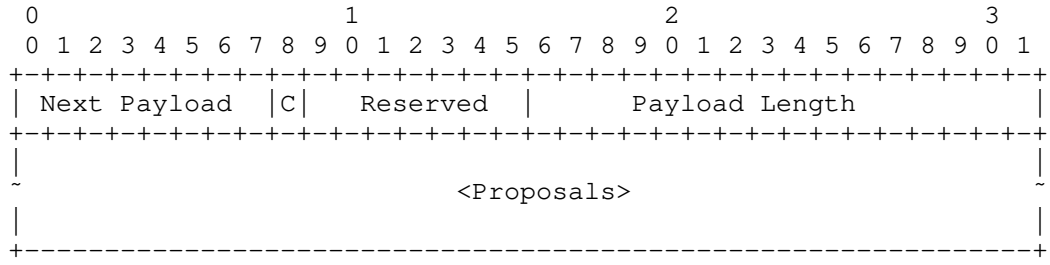


Figure 8: Security Association Payload

Proposals (Variable) - one or more proposal substructures

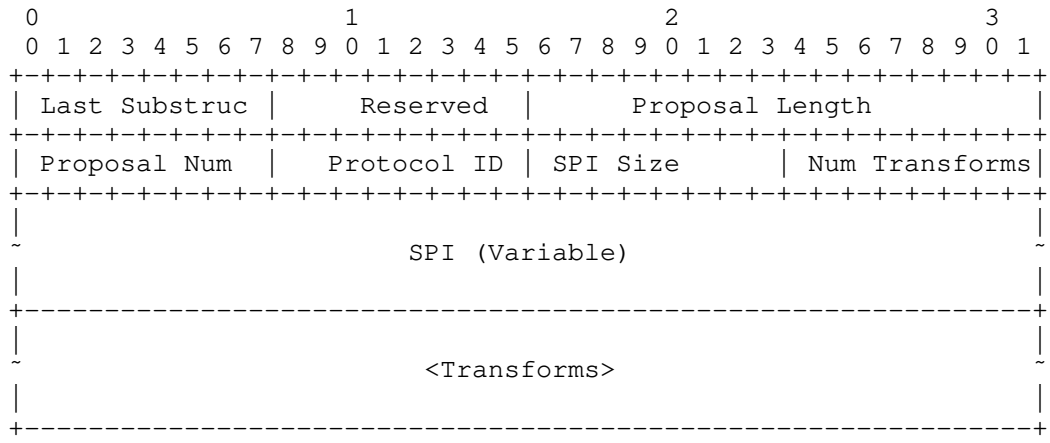


Figure 9: Proposal Substructure

6 Applicability to other VPN types

Although P2MP BGP signaling for establishment and maintenance of SAs among PE devices is described in this document in context of EVPN, there is no reason why it cannot be extended to other VPN technologies such as IP-VPN [RFC4364], VPLS [RFC4761] & [RFC4762], and MVPN [RFC6513] & [RFC6514] with ingress replication. The reason EVPN has been chosen is because of its pervasiveness in DC, SP, and Enterprise applications and because of its ability to support SA establishment at different granularity levels such as: per PE, Per tenant, per subnet, per Ethernet Segment, per IP address, and per MAC. For other VPN technology types, a much smaller granularity levels can be supported. For example for VPLS, only the granularity of per PE and per subnet can be supported. For per-PE granularity level, the mechanism is the same among all the VPN technologies as IPsec tunnel type (and its associated TLV and sub-TLVs) are sent along with the PE's loopback IPv4 (or IPv6) address. For VPLS, if per-subnet (per bridge domain) granularity level needs to be supported, then the IPsec tunnel type and TLV are sent along with VPLS AD route.

The following table lists what level of granularity can be supported by a given VPN technology and with what BGP route.

Functionality	EVPN	IP-VPN	MVPN	VPLS
per PE	IPv4/v6 route	IPv4/v6 route	IPv4/v6 rte	IPv4/v6
per tenant	IMET (or new)	lpbk (or new)	I-PMSI	N/A
per subnet	IMET	N/A	N/A	VPLS AD
per IP	EVPN RT2/RT5	VPN IP rt	*,G or S,G	N/A
per MAC	EVPN RT2	N/A	N/A	N/A

7 Acknowledgements

8 Security Considerations

9 IANA Considerations

A new transitive extended community Type of 0x06 and Sub-Type of TBD for EVPN Attachment Circuit Extended Community needs to be allocated by IANA.

10 References

10.1 Normative References

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

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017.

[RFC7432] Sajassi et al., "BGP MPLS Based Ethernet VPN", RFC 7432, February, 2015.

[RFC8365] Sajassi et al., "A Network Virtualization Overlay Solution

Using Ethernet VPN (EVPN)", RFC 8365, March, 2018.

[TUNNEL-ENCAP] Rosen et al., "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-03, November 2016.

[CONTROLLER-IKE] Carrel et al., "IPsec Key Exchange using a Controller", draft-carrel-ipsecme-controller-ike-00, July, 2018.

[RFC3948] Huttunen et al., "UDP Encapsulation of IPsec ESP Packets", RFC 3948, January 2005.

[IKEV2-IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", February 2016, www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005.

10.2 Informative References

[RFC4364] Rosen, E., et. al., "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

[RFC4761] Kompella, K., et. al., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, January 2007.

[RFC4762] Kompella, K., et. al., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, January 2007.

[RFC6513] Rosen, E., et. al., "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.

[RFC6514] Rosen, E., et. al., "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.

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

[802.1Q] "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q(tm), 2014 Edition, November 2014.

[RFC7348] Mahalingam, M., et al., "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014.

[GENEVE] Gross, J., et al., "Geneve: Generic Network Virtualization Encapsulation", Work in Progress, draft-ietf-nvo3-geneve-06, March 2018.

Authors' Addresses

Ali Sajassi
Cisco
Email: sajassi@cisco.com

Ayan Banerjee
Cisco
Email: ayabaner@cisco.com

Samir Thoria
Cisco
Email: sthoria@cisco.com

David Carrel
Cisco
Email: carrel@cisco.com

Brian Weis
Cisco
Email: bew@cisco.com

BESS Workgroup
INTERNET-DRAFT
Intended Status: Standards Track

A. Sajassi, Ed.
A. Banerjee
S. Thoria
D. Carrel
Cisco
B. Weis
Individual
J. Drake
Juniper

Expires: January 13, 2021

July 13, 2020

Secure EVPN
draft-sajassi-bess-secure-evpn-03

Abstract

The applications of EVPN-based solutions ([RFC7432] and [RFC8365]) have become pervasive in Data Center, Service Provider, and Enterprise segments. It is being used for fabric overlays and inter-site connectivity in the Data Center market segment, for Layer-2, Layer-3, and IRB VPN services in the Service Provider market segment, and for fabric overlay and WAN connectivity in Enterprise networks. For Data Center and Enterprise applications, there is a need to provide inter-site and WAN connectivity over public Internet in a secured manner with same level of privacy, integrity, and authentication for tenant's traffic as IPsec tunneling using IKEv2. This document presents a solution where BGP point-to-multipoint signaling is leveraged for key and policy exchange among PE devices to create private pair-wise IPsec Security Associations without IKEv2 point-to-point signaling or any other direct peer-to-peer session establishment messages.

Status of this Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months

Copyright and License Notice

Table of Contents

Terminology

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

AC: Attachment Circuit.

ARP: Address Resolution Protocol.

BD: Broadcast Domain. As per [RFC7432], an EVI consists of a single or multiple BDs. In case of VLAN-bundle and VLAN-based service models (see [RFC7432]), a BD is equivalent to an EVI. In case of VLAN-aware bundle service model, an EVI contains multiple BDs. Also, in this document, BD and subnet are equivalent terms.

BD Route Target: refers to the Broadcast Domain assigned Route Target [RFC4364]. In case of VLAN-aware bundle service model, all the BD instances in the MAC-VRF share the same Route Target.

BT: Bridge Table. The instantiation of a BD in a MAC-VRF, as per [RFC7432].

DGW: Data Center Gateway.

Ethernet A-D route: Ethernet Auto-Discovery (A-D) route, as per [RFC7432].

Ethernet NVO tunnel: refers to Network Virtualization Overlay tunnels with Ethernet payload. Examples of this type of tunnels are VXLAN or GENEVE.

EVI: EVPN Instance spanning the NVE/PE devices that are participating on that EVPN, as per [RFC7432].

EVPN: Ethernet Virtual Private Networks, as per [RFC7432].

GRE: Generic Routing Encapsulation.

GW IP: Gateway IP Address.

IPL: IP Prefix Length.

IP NVO tunnel: it refers to Network Virtualization Overlay tunnels with IP payload (no MAC header in the payload).

IP-VRF: A VPN Routing and Forwarding table for IP routes on an NVE/PE. The IP routes could be populated by EVPN and IP-VPN address families. An IP-VRF is also an instantiation of a layer 3 VPN in an NVE/PE.

IRB: Integrated Routing and Bridging interface. It connects an IP-VRF to a BD (or subnet).

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on an NVE/PE, as per [RFC7432]. A MAC-VRF is also an instantiation of an EVI in an NVE/PE.

ML: MAC address length.

ND: Neighbor Discovery Protocol.

NVE: Network Virtualization Edge.

GENEVE: Generic Network Virtualization Encapsulation, [GENEVE].

NVO: Network Virtualization Overlays.

RT-2: EVPN route type 2, i.e., MAC/IP advertisement route, as defined in [RFC7432].

RT-5: EVPN route type 5, i.e., IP Prefix route. As defined in Section 3 of [EVPN-PREFIX].

SBD: Supplementary Broadcast Domain. A BD that does not have any ACs, only IRB interfaces, and it is used to provide connectivity among all the IP-VRFs of the tenant. The SBD is only required in IP-VRF- to-IP-VRF use-cases (see Section 4.4.).

SN: Subnet.

TS: Tenant System.

VA: Virtual Appliance.

VNI: Virtual Network Identifier. As in [RFC8365], the term is used as a representation of a 24-bit NVO instance identifier, with the understanding that VNI will refer to a VXLAN Network Identifier in VXLAN, or Virtual Network Identifier in GENEVE, etc. unless it is stated otherwise.

VTEP: VXLAN Termination End Point, as in [RFC7348].

VXLAN: Virtual Extensible LAN, as in [RFC7348].

This document also assumes familiarity with the terminology of [RFC7432], [RFC8365] and [RFC7365].

1 Introduction

The applications of EVPN-based solutions have become pervasive in Data Center, Service Provider, and Enterprise segments. It is being used for fabric overlays and inter-site connectivity in the Data Center market segment, for Layer-2, Layer-3, and IRB VPN services in the Service Provider market segment, and for fabric overlay and WAN connectivity in the Enterprise networks. For Data Center and Enterprise applications, there is a need to provide inter-site and WAN connectivity over public Internet in a secured manner with the same level of privacy, integrity, and authentication for tenant's traffic as used in IPsec tunneling using IKEv2. This document presents a solution where BGP point-to-multipoint signaling is leveraged for key and policy exchange among PE devices to create private pair-wise IPsec Security Associations without IKEv2 point-to-point signaling or any other direct peer-to-peer session establishment messages.

EVPN uses BGP as control-plane protocol for distribution of information needed for discovery of PEs participating in a VPN, discovery of PEs participating in a redundancy group, customer MAC addresses and IP prefixes/addresses, aliasing information, tunnel encapsulation types, multicast tunnel types, multicast group memberships, and other info. The advantages of using BGP control plane in EVPN are well understood including the following:

1) A full mesh of BGP sessions among PE devices can be avoided by using Route Reflector (RR) where a PE only needs to setup a single BGP session between itself and the RR as opposed to setting up N BGP sessions to N other remote PEs; therefore, reducing number of BGP sessions from O(N^2) to O(N) in the network. Furthermore, RR hierarchy can be leveraged to scale the number of BGP routes on the RR.

2) MP-BGP route filtering and constrained route distribution can be leveraged to ensure that the control-plane traffic for a given VPN is only distributed to the PEs participating in that VPN.

For setting up point-to-point security association (i.e., IPsec tunnel) between a pair of EVPN PEs, it is important to leverage BGP point-to-multipoint singling architecture using the RR along with its route filtering and constrain mechanisms to achieve the performance and the scale needed for large number of security associations (IPsec tunnels) along with their frequent re-keying requirements. Using BGP signaling along with the RR (instead of peer-to-peer protocol such as IKEv2) reduces number of message exchanges needed for SAs establishment and maintenance from O(N^2) to O(N) in the network.

2 Requirements

The requirements for secured EVPN are captured in the following subsections.

2.1 Tenant's Layer-2 and Layer-3 data & control traffic

Tenant's layer-2 and layer-3 data and control traffic must be protected by IPsec cryptographic methods. This implies not only tenant's data traffic must be protected by IPsec but also tenant's control and routing information that are advertised in BGP must also be protected by IPsec. This in turn implies that BGP session must be protected by IPsec.

2.2 Tenant's Unicast & Multicast Data Protection

Tenant's layer-2 and layer-3 unicast traffic must be protected by IPsec. In addition to that, tenant's layer-2 broadcast, unknown unicast, and multicast traffic as well as tenant's layer-3 multicast traffic must be protected by IPsec when ingress replication or assisted replication are used. The use of BGP P2MP signaling for setting up P2MP SAs in P2MP multicast tunnels is for future study.

2.3 P2MP Signaling for SA setup and Maintenance

BGP P2MP signaling must be used for IPsec SAs setup and maintenance. The BGP signaling must follow P2MP signaling framework per [CONTROLLER-IKE] for IPsec SAs setup and maintenance in order to reduce the number of message exchanges from O(N^2) to O(N) among the participant PE devices.

2.4 Granularity of Security Association Tunnels

The solution must support the setup and maintenance of IPsec SAs at the following level of granularities:

1) Per PE: A single IPsec tunnel between a pair of PEs to be used for all tenants' traffic supported by the pair of PEs.

2) Per tenant: A single IPsec tunnel per tenant per pair of PEs. For example, if there are 1000 tenants supported on a pair of PEs, then 1000 IPsec tunnels are required between that pair of PEs.

3) Per subnet: A single IPsec tunnel per subnet (e.g., per VLAN/EVI) of a tenant on a pair of PEs.

4) Per IP address: A single IPsec tunnel per pair of IP addresses of a tenant on a pair of PEs.

5) Per MAC address: A single IPsec tunnel per pair of MAC addresses of a tenant on a pair of PEs.

6) Per Attachment Circuit: A single IPsec tunnel per pair of Attachment Circuits between a pair of PEs.

2.5 Support for Policy and DH-Group List

The solution must support a single policy and DH group for all SAs as well as supporting multiple policies and DH groups among the SAs.

3 BGP Component

The architecture that encompasses device-to-controller trust model, has several components among which is the signaling component. Secure EVPN Signaling, as defined in this document, is the BGP signaling component of the overall Architecture. We will briefly describe this Architecture here to further facilitate understanding how Secure EVPN fits into the overall architecture. The Architecture describes the components needed to create BGP based SD-WANs and how these components work together. Our intention is to list these components here along with their brief description and to describe this Architecture in details in a separate document where to specify the details for other parts of this architecture besides the BGP signaling component which is described in this document.

The Architecture consists of four components. These components are Zero Touch Bring-up, Configuration Management, Orchestration, and Signaling. In addition to these components, secure communications must be provided between the edge nodes and all servers/devices providing the architecture components.

3.1 Zero Touch Bring-up (ZTB)

The first component is a zero touch capability that allows an edge device to find and join its SD-WAN with little to no assistance other than power and network connectivity. The goal is to use existing work in this area. The requirements are that an edge device can locate its ZTB server/component of its SD-WAN controller in a secure manner and to proceed to receive its configuration.

3.2 Configuration Management

After an edge device joins its SD-WAN, it needs to be configured.

Configuration covers all device configuration, not just the configuration related to Secure EVPN. The previous Zero Touch Bring-up component will have directed the edge device, either directly or indirectly, to its configuration server/component. One example of a configuration server is the I2NSF Controller. After a device has been configured, it can engage in the next two components. Configuration may include updates over time and is not a one time only component.

3.3 Orchestration

This component is optional. It allows for more dynamic updates of configuration and statistics information. Orchestration can be more dynamic than configuration.

3.4 Signaling

Signaling is the component described in this document. The functionality of a Route Reflector is well understood. Here we describe the signaling component of BGP SD-WAN Architecture and the BGP extension/signaling for IPsec key management and policy.

4 Solution Description

This solution uses BGP P2MP signaling where an originating PE only send a message to the Route Reflector (RR) and then the RR reflects that message to the interested recipient PEs. The framework for such signaling is described in [CONTROLLER-IKE] and it is referred to as device-to-controller trust model. This trust model is significantly different than the traditional peer-to-peer trust model where a P2P signaling protocol such as IKEv2 [RFC7296] is used in which the PE devices directly authenticate each other and agree upon security policy and keying material to protect communications between themselves. The device-to-controller trust model leverages P2MP signaling via the controller (e.g., the RR) to achieve much better scale and performance for establishment and maintenance of large number of pair-wise Security Associations (SAs) among the PEs.

This device-to-controller trust model first secures the control channel between each device and the controller using peer-to-peer protocol such as IKEv2 [RFC7296] to establish P2P SAs between each PE and the RR. It then uses this secured control channel for P2MP signaling in establishment of P2P SAs between each pair of PE devices.

Each PE advertises to other PEs via the RR the information needed in establishment of pair-wise SAs between itself an every other remote PEs. These pieces of information are sent as Sub-TLVs of IPSec tunnel type in BGP Tunnel Encapsulation attribute. These Sub-TLVs are detailed in section 5 and are based on the DIM message components from [CONTROLLER-IKE] and the IKEv2 specification [RFC7296]. The IPsec tunnel TLVs along with its Sub-TLVs are sent along with the BGP route (NLRI) for a given level of granularity.

If only a single SA is required per pair of PE devices to multiplex user traffic for all tenants, then IPsec tunnel TLV is advertised along with IPv4 or IPv6 NLRI representing loopback address of the originating PE. It should be noted that this is not a VPN route but rather an IPv4 or IPv6 route.

If a SA is required per tenant between a pair of PE devices, then IPsec tunnel TLV can be advertised along with EVPN IMET route representing the tenant or can be advertised along with a new EVPN route representing the tenant.

If a SA is required per tenant's subnet (e.g., per VLAN) between a pair of PE devices, then IPsec tunnel TLV is advertised along with EVPN IMET route.

If a SA is required between a pair of tenant's devices represented by a pair of IP addresses, then IPsec tunnel TLV is advertised along with EVPN IP Prefix Advertisement Route or EVPN MAC/IP Advertisement route.

If a SA is required between a pair of tenant's devices represented by a pair of MAC addresses, then IPsec tunnel TLV is advertised along with EVPN MAC/IP Advertisement route.

If a SA is required between a pair of Attachment Circuits (ACs) on two PE devices (where an AC can be represented by <VLAN, port>), then IPsec tunnel TLV is advertised along with EVPN Ethernet AD route.

4.1 Inheritance of Security Policies

Operationally, it is easy to configure a security association between a pair of PEs using BGP signaling. This is the default security association that is used for traffic that flows between peers. However, in the event more finer granularity of security association is desired on the traffic flows, it is possible to set up SAs between a pair of tenants, a pair of subnets within a tenant, a pair of IPs between a subnet, and a pair of MACs between a subnet using the appropriate EVPN routes as described above. In the event, there are no security TLVs associated with an EVPN route, there is a strict

order in the manner security associations are inherited for such a route. This results in an EVPN route inheriting the security associations of the parent in a hierarchical fashion. For example, traffic between an IP pair is protected using security TLVs announced along with the EVPN IP Prefix Advertisement Route or EVPN MAC/IP Advertisement route as a first choice. If such TLVs are missing with the associated route, then one checks to see if the subnets the IPs are associated with has security TLVs with the EVPN IMET route. If they are present, those associations are used in securing the traffic. In the absence of them, the peer security associations are used. The order in which security associations are inherited are from the granular to the coarser, namely, IP/MAC associated TLVs with the EVPN route being the first preference, and the subnet, the tenant, and the peer associations preferred in that fashion.

It should be noted that when a security association is made it is possible for it to be re-used by a large number of traffic flows. For example, a tenant security association may be associated with a number of child subnet routes. Clearly it is mandatory to keep a tenant security association alive, if there are one or more subnet routes that want to use that association. Logically, the security associations between a pair of entities creates a single secure tunnel. It is thus possible to classify the incoming traffic in the most granular sense {IP/MAC, subnet, tenant, peer} to a particular secure tunnel that falls within its route hierarchy. The policy that is applied to such traffic is independent from its use of an existing or a new secure tunnel. It is clear that since any number of classified traffic flows can use a security association, such a security association will not be torn down, if at least there is one policy using such a secure tunnel.

4.2 Distribution of Public Keys and Policies

One of the requirements for this solution is to support a single DH group and a single policy for all SAs as well as to support multiple DH groups and policies among the SAs. The following subsections describe what pieces of information (what Sub-TLVs) are needed to be exchanged to support a single DH group and a single policy versus multiple DH groups and multiple policies.

4.2.1 Minimal DIM

For SA establishment, at the minimum, a PE needs to advertise to other PEs, its DIM values as specified in [CONTROLLER-IKE]. These include:

ID Tunnel ID
N Nonce

RC Rekey Counter
I Indication of initial policy distribution
KE DH public value.

When this minimal set of DIM values is sent, then it is assumed that all peer PEs share the same policy for which DH group to use, as well as which IPSec SA policy to employ. Section 5.1 defines the Minimal DIM sub-TLV as part of IPsec tunnel TLV in BGP Tunnel Encapsulation Attribute.

4.2.2 Multiple Policies

There can be scenarios for which there is a need to have multiple policy options. This can happen when there is a need for policy change and smooth migration among all PE devices to the new policy is required. It can also happen if different PE devices have different capabilities within the network. In these scenarios, PE devices need to be able to choose the correct policy to use for each other. This multi-policy scheme is described in section 6 of [CONTROLLER-IKE]. In order to support this multi-policy feature, a PE device MUST distribute a policy list. This list consists of multiple distinct policies in order of preference, where the first policy is the most preferred one. The receiving PE selects the policy by taking the received list (starting with the first policy) and comparing that against its own list and choosing the first one found in common. If there is no match, this indicates a configuration error and the PEs MUST NOT establish new SAs until a message is received that does produce a match.

4.2.2.1 Multiple DH-groups

It can be the case that not all peers use the same DH group. When multiple DH groups are supported, the peer may include multiple KE Sub-TLVs. The order of the KE Sub-TLVs determines the preference. The preference and selection methods are specified in Section 6 of [CONTROLLER-IKE].

4.2.2.2 Multiple or Single ESP SA policies

In order to specify an ESP SA Policy, a DIM may include one or more SA Sub-TLVs. When all peers are configured by a controller with the same ESP SA policy, they MAY leave the SA out of the DIM. This minimizes messaging when group configuration is static and known. However, it may also be desirable to include the SA. If a single SA is included, the peer is indicating what ESP SA policy it uses, but is not willing to negotiate. If multiple SA Sub-TLVs are included, the peer is indicating that it is willing to negotiate. The order of

the SA Sub-TLVs determines the preference. The preference and selection methods are specified in Section 6 of [CONTROLLER-IKE].

4.3 Initial IPsec SAs Generation

The procedure for generation of initial IPsec SAs is described in section 3 of [CONTROLLER-IKE]. This section gives a summary of it in context of BGP signaling. When a PE device first comes up and wants to setup an IPsec SA between itself and each of the interested remote PEs, it generates a DH pair along for each [what word here? "tennant"?] using an algorithm defined in the IKEv2 Diffie-Hellman Group Transform IDs [IKEv2-IANA]. The originating PE distributes the DH public value along with the other values in the DIM (using IPsec Tunnel TLV in Tunnel Encapsulation Attribute) to other remote PEs via the RR. Each receiving PE uses this DH public number and the corresponding nonce in creation of IPsec SA pair to the originating PE - i.e., an outbound SA and an inbound SA. The detail procedures are described in section 5.2 of [CONTROLLER-IKE].

4.4 Re-Keying

A PE can initiate re-keying at any time due to local time or volume based policy or due to the result of cipher counter nearing its final value. The rekey process is performed individually for each remote PE. If rekeying is performed with multiple PEs simultaneously, then the decision process and rules described in this rekey are performed independently for each PE. Section 4 of [CONTROLLER-IKE] describes this rekeying process in details and gives examples for a single IPsec device (e.g., a single PE) rekey versus multiple PE devices rekey simultaneously.

4.5 IPsec Databases

The Peer Authorization Database (PAD), the Security Policy Database (SPD), and the Security Association Database (SAD) all need to be setup as defined in the IPsec Security Architecture [RFC4301]. Section 5 of [CONTROLLER-IKE] gives a summary description of how these databases are setup for the controller-based model where key is exchanged via P2MP signaling via the controller (i.e., the RR) and the policy can be either signaled via the RR (in case of multiple policies) or configured by the management station (in case of single policy).

5 Encapsulation

Vast majority of Encapsulation for Network Virtualization Overlay (NVO) networks in deployment are based on UDP/IP with UDP destination port ID indicating the type of NVO encapsulation (e.g., VxLAN, GPE, GENEVE, GUE) and UDP source port ID representing flow entropy for load-balancing of the traffic within the fabric based on n-tuple that includes UDP header. When encrypting NVO encapsulated packets using IP Encapsulating Security Payload (ESP), the following two options can be used: a) adding a UDP header before ESP header (e.g., UDP header in clear) and b) no UDP header before ESP header (e.g., standard ESP encapsulation). The following subsection describe these encapsulation in further details.

5.1 Standard ESP Encapsulation

When standard IP Encapsulating Security Payload (ESP) is used (without outer UDP header) for encryption of NVO packets, it is used in transport mode as depicted below. When such encapsulation is used, for BGP signaling, the Tunnel Type of Tunnel Encapsulation TLV is set to ESP-Transport and the Tunnel Type of Encapsulation Extended Community is set to NVO encapsulation type (e.g., VxLAN, GENEVE, GPE, etc.). This implies that the customer packets are first encapsulated using NVO encapsulation type and then it is further encapsulated & encrypted using ESP-Transport mode.

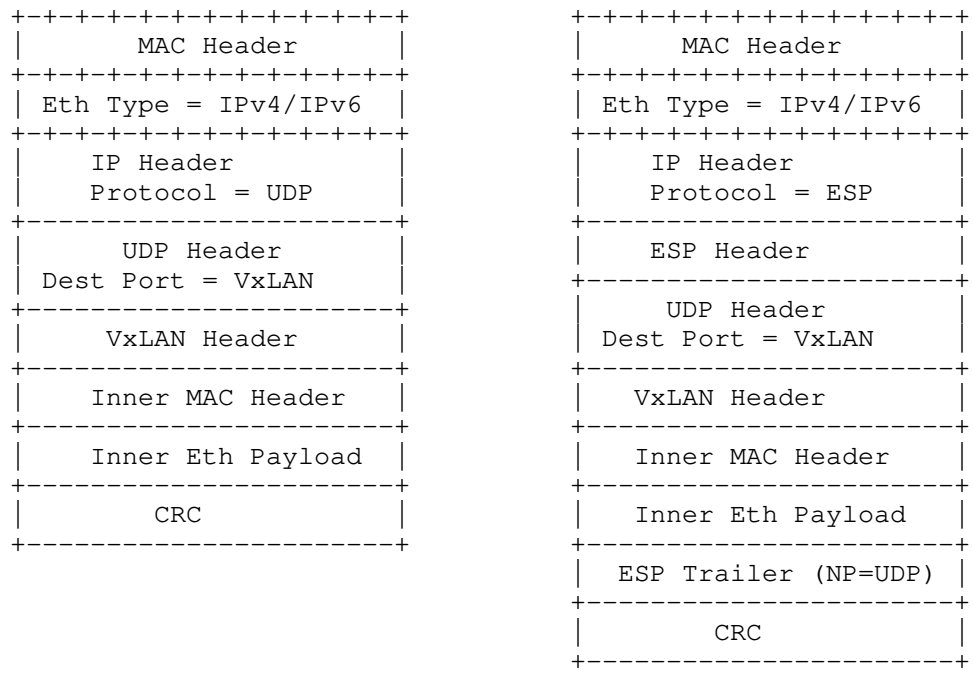


Figure 3: VxLAN Encapsulation within ESP

5.2 ESP Encapsulation within UDP packet

In scenarios where NAT traversal is required ([RFC3948]) or where load balancing using UDP header is required, then ESP encapsulation within UDP packet as depicted in the following figure is used. The ESP for NVO applications is in transport mode. The outer UDP header (before the ESP header) has its source port set to flow entropy and its destination port set to 4500 (indicating ESP header follows). A non-zero SPI value in ESP header implies that this is a data packet (i.e., it is not an IKE packet). The Next Protocol field in the ESP trailer indicates what follows the ESP header, is a UDP header. This inner UDP header has a destination port ID that identifies NVO encapsulation type (e.g., VxLAN). Optimization of this packet format where only a single UDP header is used (only the outer UDP header) is for future study.

When such encapsulation is used, for BGP signaling, the Tunnel Type of Tunnel Encapsulation TLV is set to ESP-in-UDP-Transport and the Tunnel Type of Encapsulation Extended Community is set to NVO

encapsulation type (e.g., VxLAN, GENEVE, GPE, etc.). This implies that the customer packets are first encapsulated using NVO encapsulation type and then it is further encapsulated & encrypted using ESP-in-UDP with Transport mode.

[RFC3948]

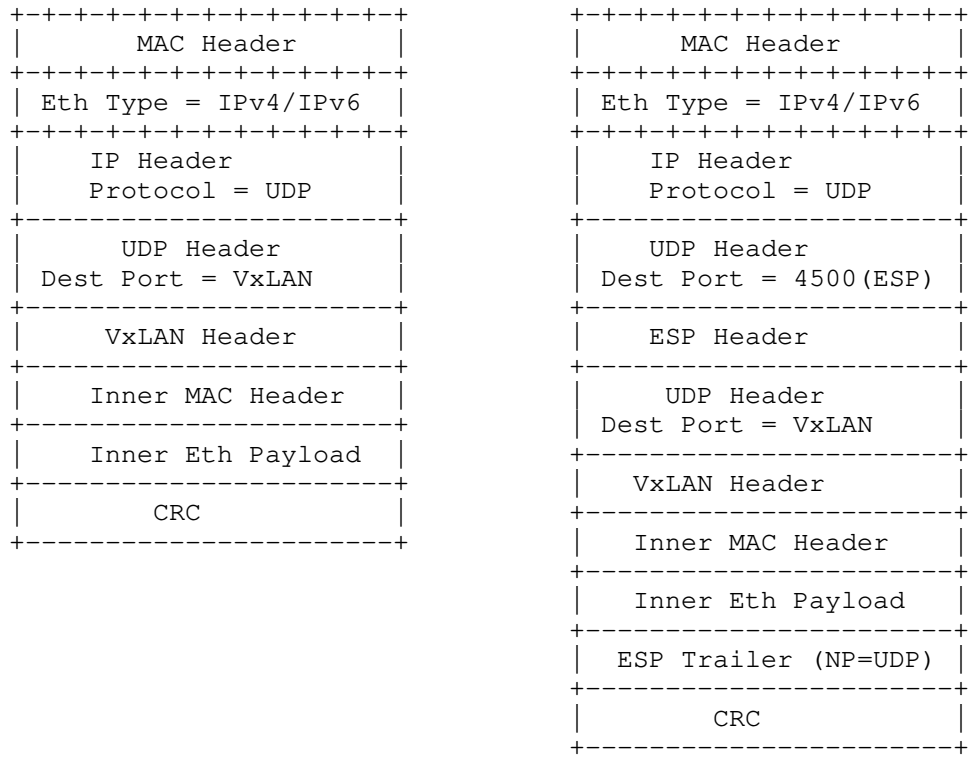


Figure 4: VxLAN Encapsulation within ESP Within UDP

6 BGP Encoding

This document defines two new Tunnel Types along with its associated sub-TLVs for The Tunnel Encapsulation Attribute [TUNNEL-ENCAP]. These tunnel types correspond to ESP-Transport and ESP-in-UDP-Transport as described in section 4. The following sub-TLVs apply to both tunnel types unless stated otherwise.

6.1 The Base (Minimal Set) DIM Sub-TLV

The Base DIM is described in 3.2.1. One and only one Base DIM may be sent in the IPSec Tunnel TLV.

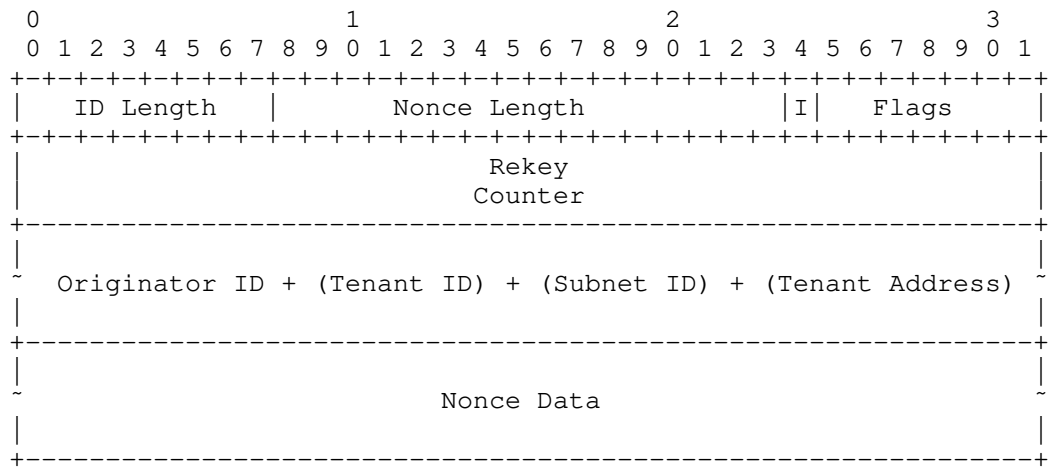


Figure 5: The Base DIM Sub-TLV

ID Length (16 bits) is the length of the Originator ID + (Tenant ID) + (Subnet ID) + (Tenant Address) in bytes.

Nonce Length (8 bits) is the length of the Nonce Data in bytes

I (1 bit) is the initial contact flag from [CONTROLLER-IKE]

Flags (7 bits) are reserved and MUST be set to zero on transmit and ignored on receipt.

The Rekey Counter is a 64 bit rekey counter as specified in [CONTROLLER-IKE]

The Originator ID + (Tenant ID) + (Subnet ID) + (Tenant Address) is the tunnel identifier and uniquely identifies the tunnel. Depending on the granularity of the tunnel, the fields in () may not be used - i.e., for a tunnel at the PE level of granularity, only Originator ID is required.

The Nonce Data is the nonce described in [CONTROLLER-IKE]. Its length is a multiple of 32 bits. Nonce lengths should be chosen to meet minimum requirements described in IKEv2 [RFC7296].

6.2 Key Exchange Sub-TLV

The KE Sub-TLV is described in 3.2.1 and 3.2.2.1. A KE is always required. One or more KE Sub-TLVs may be included in the IPSec Tunnel TLV.

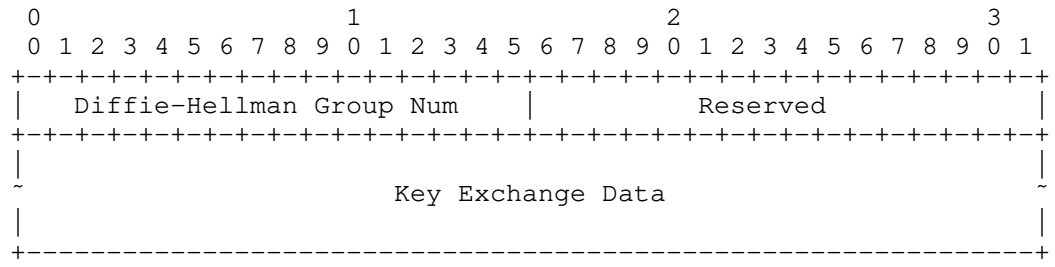


Figure 6: Key Exchange Sub-TLV

Diffie-Hellman Group Num 916 bits) identifies the Diffie-Hellman group in the Key Exchange Data was computed. Diffie-Hellman group numbers are discussed in IKEv2 [RFC7296] Appendix B and [RFC5114].

The Key Exchange payload is constructed by copying one's Diffie-Hellman public value into the "Key Exchange Data" portion of the payload. The length of the Diffie-Hellman public value is described for MOPD groups in [RFC7296] and for ECP groups in [RFC4753].

6.3 ESP SA Proposals Sub-TLV

The SA Sub-TLV is described in 3.2.2.2. Zero or more SA Sub-TLVs may be included in the IPSec Tunnel TLV.

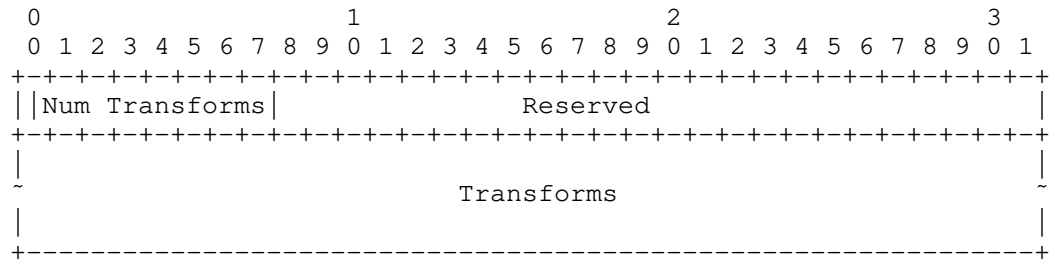


Figure 8: ESP SA Proposals Sub-TLV

Num Transforms is the number of transforms included.

Reserved is not used and MUST be set to zero on transmit and MUST be ignored on receipt.

6.3.1 Transform Substructure

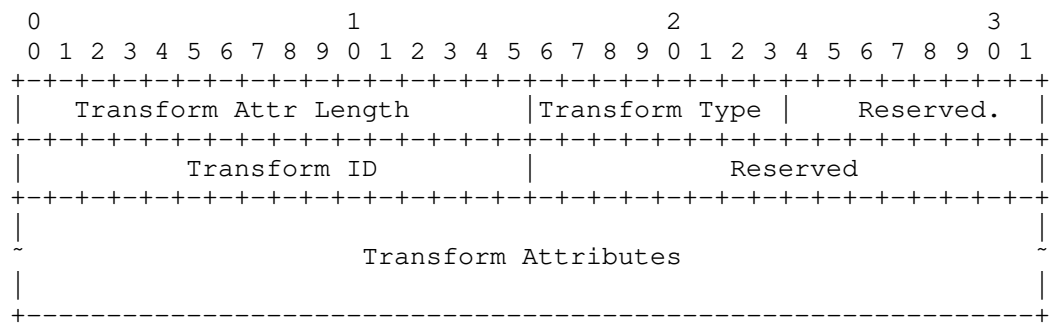


Figure 9: Transform Substructure Sub-TLV

The Transform Attr Length is the length of the Transform Attributes field.

The Transform Type is from Section 3.3.2 of [RFC7296] and [IKEV2IANA]. Only the values ENCR, INTEG, and ESN are allowed.

The Transform ID specifies the transform identification value from [IKEV2IANA].

Reserved is unused and MUST be zero on transmit and MUST be ignored on receipt.

The Transform Attributes are taken directly from 3.3.5 of [RFC7296].

7 Applicability to other VPN types

Although P2MP BGP signaling for establishment and maintenance of SAs among PE devices is described in this document in context of EVPN, there is no reason why it cannot be extended to other VPN technologies such as IP-VPN [RFC4364], VPLS [RFC4761] & [RFC4762], and MVPN [RFC6513] & [RFC6514] with ingress replication. The reason EVPN has been chosen is because of its pervasiveness in DC, SP, and Enterprise applications and because of its ability to support SA establishment at different granularity levels such as: per PE, Per tenant, per subnet, per Ethernet Segment, per IP address, and per MAC. For other VPN technology types, a much smaller granularity levels can be supported. For example for VPLS, only the granularity of per PE and per subnet can be supported. For per-PE granularity level, the mechanism is the same among all the VPN technologies as IPsec tunnel type (and its associated TLV and sub-TLVs) are sent along with the PE's loopback IPv4 (or IPv6) address. For VPLS, if

per-subnet (per bridge domain) granularity level needs to be supported, then the IPsec tunnel type and TLV are sent along with VPLS AD route.

The following table lists what level of granularity can be supported by a given VPN technology and with what BGP route.

Functionality	EVPN	IP-VPN	MVPN	VPLS
per PE	IPv4/v6 route	IPv4/v6 route	IPv4/v6 rte	IPv4/v6
per tenant	IMET (or new)	lpbk (or new)	I-PMSI	N/A
per subnet	IMET	N/A	N/A	VPLS AD
per IP	EVPN RT2/RT5	VPN IP rt	*,G or S,G	N/A
per MAC	EVPN RT2	N/A	N/A	N/A

8 Acknowledgements

9 Security Considerations

10 IANA Considerations

A new transitive extended community Type of 0x06 and Sub-Type of TBD for EVPN Attachment Circuit Extended Community needs to be allocated by IANA.

10 References

11.1 Normative References

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

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC2119

Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017.

[RFC7432] Sajassi et al., "BGP MPLS Based Ethernet VPN", RFC 7432, February, 2015.

[RFC8365] Sajassi et al., "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, March, 2018.

[TUNNEL-ENCAP] Rosen et al., "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-03, November 2016.

[CONTROLLER-IKE] Carrel et al., "IPsec Key Exchange using a Controller", draft-carrel-ipsecme-controller-ike-00, July, 2018.

[IKEV2IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", <http://www.iana.org/assignments/ikev2-parameters/>.

[RFC3948] Huttunen et al., "UDP Encapsulation of IPsec ESP Packets", RFC 3948, January 2005.

[IKEV2-IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", February 2016, www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005.

11.2 Informative References

[RFC4364] Rosen, E., et. al., "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

[RFC4761] Kompella, K., et. al., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, January 2007.

[RFC4762] Kompella, K., et. al., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, January 2007.

[RFC6513] Rosen, E., et. al., "Multicast in MPLS/BGP IP VPNs", RFC

6513, February 2012.

[RFC6514] Rosen, E., et. al., "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.

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

[802.1Q] "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q(tm), 2014 Edition, November 2014.

[RFC7348] Mahalingam, M., et al., "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014.

[GENEVE] Gross, J., et al., "Geneve: Generic Network Virtualization Encapsulation", Work in Progress, draft-ietf-nvo3-geneve-06, March 2018.

Authors' Addresses

Ali Sajassi
Cisco
Email: sajassi@cisco.com

Ayan Banerjee
Cisco
Email: ayabaner@cisco.com

Samir Thoria
Cisco
Email: sthoria@cisco.com

David Carrel
Cisco
Email: carrel@cisco.com

Brian Weis
Individual

Email: bew.stds@gmail.com

John Drake
Juniper
Email: jdrake@juniper.net

INTERNET-DRAFT
Intended Status: Informational

Samer Salam
Ali Sajassi
Cisco
Sam Aldrin
Google
John E. Drake
Juniper
Donald Eastlake
Huawei

Expires: April 21, 2018 October 22, 2018

EVPN Operations, Administration and Maintenance Requirements and Framework
draft-salam-bess-evpn-oam-req-frmwk-01

Abstract

This document specifies the requirements and reference framework for Ethernet VPN (EVPN) Operations, Administration and Maintenance (OAM). The requirements cover the OAM aspects of EVPN and PBB-EVPN. The framework defines the layered OAM model encompassing the EVPN service layer, network layer and underlying Packet Switched Network (PSN) transport layer.

Status of this Memo

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

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

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

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html. The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Copyright and License Notice

Table of Contents

1. Introduction

This document specifies the requirements and defines a reference framework for Ethernet VPN (EVPN) Operations, Administration and Maintenance (OAM, [RFC6291]). In this context, we use the term EVPN OAM to loosely refer to the OAM functions required for and/or applicable to [RFC7432] and [RFC7623].

EVPN is an Layer 2 VPN (L2VPN) solution for multipoint Ethernet services, with advanced multi-homing capabilities, using BGP for distributing customer/client MAC address reachability information over the core MPLS/IP network.

PBB-EVPN combines Provider Backbone Bridging (PBB) [802.1Q] with EVPN in order to reduce the number of BGP MAC advertisement routes, provide client MAC address mobility using C-MAC aggregation and B-MAC sub-netting, confine the scope of C-MAC learning to only active flows, offer per site policies, and avoid C-MAC address flushing on topology changes.

This document focuses on the fault management and performance management aspects of EVPN OAM.

1.1 Relationship to Other OAM Work

This document leverages concepts and draws upon elements defined and/or used in the following documents:

[RFC6136] specifies the requirements and a reference model for OAM as it relates to L2VPN services, pseudowires and associated Packet Switched Network (PSN) tunnels. This document focuses on VPLS and VPWS solutions and services.

[RFC8029] defines mechanisms for detecting data plane failures in MPLS LSPs, including procedures to check the correct operation of the data plane, as well as mechanisms to verify the data plane against the control plane.

[802.1Q] specifies the Ethernet Connectivity Fault Management (CFM) protocol, which defines the concepts of Maintenance Domains, Maintenance Associations, Maintenance End Points, and Maintenance Intermediate Points.

[Y.1731] extends Connectivity Fault Management in the following areas: it defines fault notification and alarm suppression functions for Ethernet. It also specifies mechanisms for Ethernet performance management, including loss, delay, jitter, and throughput measurement.

1.2 Specification of Requirements

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

1.3 Terminology

This document uses the following terminology defined in [RFC6136]:

MA Maintenance Association is a set of MEPs belonging to the same Maintenance Domain, established to verify the integrity of a single service instance.

MEP Maintenance End Point is responsible for origination and termination of OAM frames for a given MA.

MIP Maintenance Intermediate Point is located between peer MEPs and can process and respond to certain OAM frames but does not initiate them.

MD Maintenance Domain, an OAM Domain that represents a region over which OAM frames can operate unobstructed.

2. EVPN OAM Framework

2.1 OAM Layering

Multiple layers come into play for implementing an L2VPN service using the EVPN family of solutions:

- The Service Layer runs end to end between the sites or Ethernet Segments that are being interconnected by the EVPN solution.

- The Network Layer extends in between the EVPN PE nodes and is mostly transparent to the core nodes (except where Flow Entropy comes into play). It leverages MPLS for service (i.e. EVI) multiplexing and Split-Horizon functions.

- The Transport Layer is dictated by the networking technology of the PSN. It may be either based on MPLS LSPs or IP.

- The Link Layer is dependent upon the physical technology used. Ethernet is a popular choice for this layer, but other alternatives are deployed (e.g. POS, DWDM etc.).

This layering extends to the set of OAM protocols that are involved in the ongoing maintenance and diagnostics of EVPN networks. The figure below depicts the OAM layering, and shows which devices have visibility into what OAM layer(s).

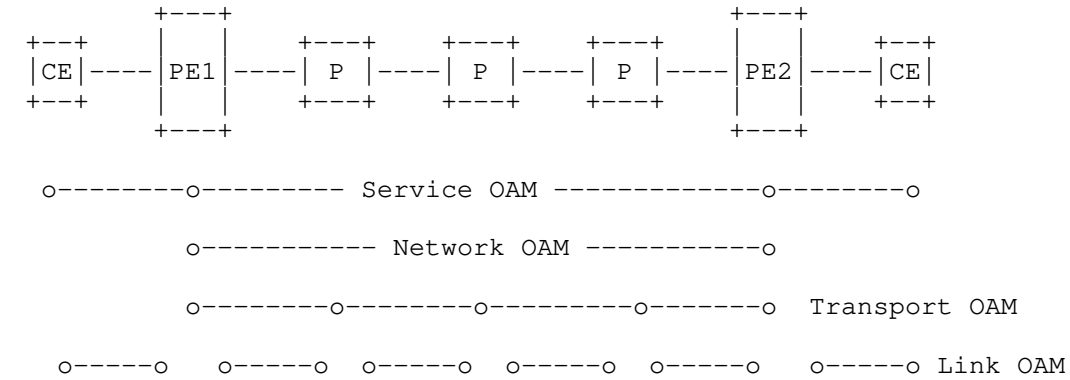


Figure 1: OAM Layering

Figure 2 below shows an example network where native Ethernet domains are interconnected via EVPN, and the OAM mechanisms applicable at each layer. The details of the layers are described in the sections below.

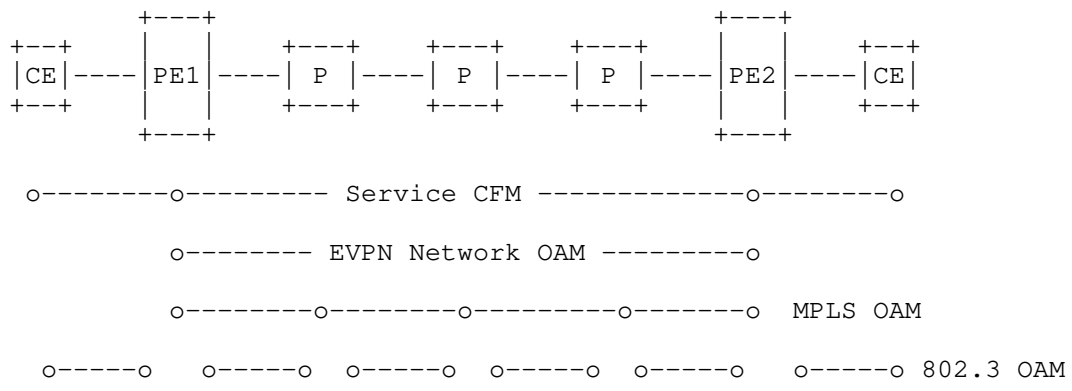


Figure 2: EVPN OAM Example

2.2 EVPN Service OAM

The EVPN Service OAM protocol depends on what service layer technology is being interconnected by the EVPN solution. In case of [RFC7432] and [RFC7623], the service layer is Ethernet; hence, the corresponding service OAM protocol is Ethernet Connectivity Fault Management (CFM) [802.1Q].

EVPN service OAM is visible to the CEs and EVPN PEs, but not to the core (P) nodes. This is because the PEs operate at the Ethernet MAC layer in [RFC7432] [RFC7623] whereas the P nodes do not.

The EVPN PE MUST support MIP functions in the applicable service OAM protocol, for example Ethernet CFM. The EVPN PE SHOULD support MEP functions in the applicable service OAM protocol. This includes both Up and Down MEP functions.

The EVPN PE MUST learn the MAC address of locally attached CE MEPs by snooping on CFM frames and advertising them to remote PEs as a MAC/IP Advertisement route.

The EVPN PE SHOULD advertise any MEP/MIP local to the PE as a MAC/IP Advertisement route. Since these are not subject to mobility, they SHOULD be advertised with the stick bit set (see Section 15.2 of [RFC7432]).

2.3 EVPN Network OAM

EVPN Network OAM is visible to the PE nodes only. This OAM layer is analogous to VCCV [RFC5085] in the case of VPLS/VPWS. It provides

mechanisms to check the correct operation of the data plane, as well as a mechanism to verify the data plane against the control plane. This includes the ability to perform fault detection and diagnostics on:

- the MP2P tunnels used for the transport of unicast traffic between PEs. EVPN allows for three different models of unicast label assignment: label per EVI, label per <ESI, Ethernet Tag> and label per MAC address. In all three models, the label is bound to an EVPN Unicast FEC.

EVPN Network OAM MUST provide mechanisms to check the operation of the data plane and verify that operation against the control plane view.

- the MP2P tunnels used for aliasing unicast traffic destined to a multi-homed Ethernet Segment. The three label assignment models, discussed above, apply here as well. In all three models, the label is bound to an EVPN Aliasing FEC. EVPN Network OAM MUST provide mechanisms to check the operation of the data plane and verify that operation against the control plane view.

- the multicast tunnels (either MP2P or P2MP) used for the transport of broadcast, unknown unicast and multicast traffic between PEs. In the case of ingress replication, a label is allocated per EVI or per <EVI, Ethernet Tag> and is bound to an EVPN Multicast FEC. In the case of LSM, and more specifically aggregate inclusive trees, again a label may be allocated per EVI or per <EVI, Ethernet Tag> and is bound to the tunnel FEC.

- the correct operation of the ESI split-horizon filtering function. In EVPN, a label is allocated per multi-homed Ethernet Segment for the purpose of performing the access split-horizon enforcement. The label is bound to an EVPN Ethernet Segment.

- the correct operation of the DF filtering function.

EVPN Network OAM MUST provide mechanisms to check the operation of the data plane and verify that operation against the control plane view for the DF filtering function.

EVPN network OAM mechanisms MUST provide in-band management capabilities. As such, OAM messages MUST be encoded so that they exhibit identical entropy characteristics to data traffic.

EVPN network OAM SHOULD provide both proactive and on-demand mechanisms of monitoring the data plane operation and data plane conformance to the state of the control plane.

2.4 Transport OAM for EVPN

The transport OAM protocol depends on the nature of the underlying transport technology in the PSN. MPLS OAM mechanisms [RFC8029] [RFC6425] as well as ICMP [RFC792] are applicable, depending on whether the PSN employs MPLS or IP transport, respectively. Furthermore, BFD mechanisms per [RFC5880], [RFC5881], [RFC5883] and [RFC5884] apply. Also, the BFD mechanisms pertaining to MPLS-TP LSPs per [RFC6428] are applicable.

2.5 Link OAM

Link OAM depends on the data link technology being used between the PE and P nodes. For example, if Ethernet links are employed, then Ethernet Link OAM [802.3] Clause 57 may be used.

2.6 OAM Inter-working

When inter-working two networking domains, such as native Ethernet and EVPN to provide an end-to-end emulated service, there is a need to identify the failure domain and location, even when a PE supports both the Service OAM mechanisms and the EVPN Network OAM mechanisms. In addition, scalability constraints may not allow running proactive monitoring, such as Ethernet Continuity Check Messages (CCMs), at a PE to detect the failure of an EVI across the EVPN domain. Thus, the mapping of alarms generated upon failure detection in one domain (e.g. native Ethernet or EVPN network domain) to the other domain is needed. There are also cases where a PE may not be able to process Service OAM messages received from a remote PE over the PSN even when such messages are defined, as in the Ethernet case, thereby necessitating support for fault notification message mapping between the EVPN Network domain and the Service domain.

OAM inter-working is not limited though to scenarios involving disparate network domains. It is possible to perform OAM inter-working across different layers in the same network domain. In general, alarms generated within an OAM layer, as a result of proactive fault detection mechanisms, may be injected into its client layer OAM mechanisms. This allows the client layer OAM to trigger event-driven (i.e. asynchronous) fault notifications. For example, alarms generated by the Link OAM mechanisms may be injected into the Transport OAM layer, and alarms generated by the Transport OAM mechanism may be injected into the Network OAM mechanism, and so on.

EVPN OAM MUST support inter-working between the Network OAM and Service OAM mechanisms. EVPN OAM MAY support inter-working among

other OAM layers.

3. EVPN OAM Requirements

This section discusses the EVPN OAM requirements pertaining to Fault Management and Performance Management.

3.1 Fault Management Requirements

3.1.1 Proactive Fault Management Functions

The network operator configures proactive fault management functions to run periodically without a time bound. Certain actions, for example protection switchover or alarm indication signaling, can be associated with specific events, such as entering or clearing fault states.

3.1.1.1 Fault Detection (Continuity Check)

Proactive fault detection is performed by periodically monitoring the reachability between service endpoints, i.e. MEPs in a given MA, through the exchange of Continuity Check messages. The reachability between any two arbitrary MEPs may be monitored for:

- in-band per-flow monitoring. This enables per flow monitoring between MEPs. EVPN Network OAM MUST support fault detection with per user flow granularity. EVPN Service OAM MAY support fault detection with per user flow granularity.

- a representative path. This enables liveness check of the nodes hosting the MEPs assuming that the loss of continuity to the MEP is interpreted as a failure of the hosting node. This, however, does not conclusively indicate liveness of the path(s) taken by user data traffic. This enables node failure detection but not path failure detection, through the use of a test flow. EVPN Network OAM and Service OAM MUST support fault detection using test flows.

- all paths. For MPLS/IP networks with ECMP, monitoring of all unicast paths between MEPs (on non-adjacent nodes) may not be possible, since the per-hop ECMP hashing behavior may yield situations where it is impossible for a MEP to pick flow entropy characteristics that result in exercising the exhaustive set of ECMP paths. Monitoring of all ECMP paths between MEPs (on non-adjacent nodes) is not a requirement for EVPN OAM.

The fact that MPLS/IP networks do not enforce congruency between

unicast and multicast paths means that the proactive fault detection mechanisms for EVPN networks MUST provide procedures to monitor the unicast paths independently of the multicast paths. This applies to EVPN Service OAM and Network OAM.

3.1.1.2 Defect Indication

EVPN Service OAM MUST support event-driven defect indication upon the detection of a connectivity defect. Defect indications can be categorized into two types: forward and reverse defect indications.

3.1.1.2.1 Forward Defect Indication

This is used to signal a failure that is detected by a lower layer OAM mechanism. A server MEP (i.e. an actual or virtual MEP) transmits a Forward Defect Indication in a direction that is away from the direction of the failure (refer to Figure 3 below).

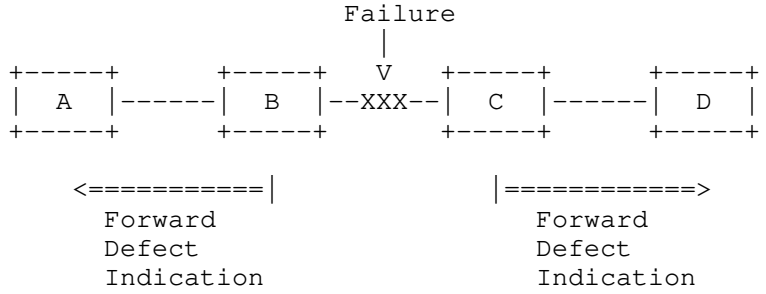


Figure 3: Forward Defect Indication

Forward defect indication may be used for alarm suppression and/or for purpose of inter-working with other layer OAM protocols. Alarm suppression is useful when a transport/network level fault translates to multiple service or flow level faults. In such a scenario, it is enough to alert a network management station (NMS) of the single transport/network level fault in lieu of flooding that NMS with a multitude of Service or Flow granularity alarms. EVPN PEs SHOULD support Forward Defect Indication in the Service OAM mechanisms.

3.1.1.2.2 Reverse Defect Indication (RDI)

RDI is used to signal that the advertising MEP has detected a loss of continuity (LoC) defect. RDI is transmitted in the direction of the

failure (refer to Figure 4).

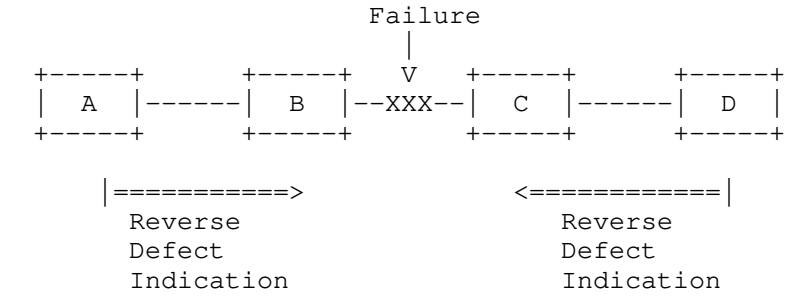


Figure 4: Reverse Defect Indication

RDI allows single-sided management, where the network operator can examine the state of a single MEP and deduce the overall health of a monitored service. EVPN PEs SHOULD support Reverse Defect Indication in the Service OAM mechanisms. This includes both the ability to signal LoC defect to a remote MEP, as well as the ability to recognize RDI from a remote MEP. Note that, in a multipoint MA, RDI is not a useful indicator of unidirectional fault. This is because RDI carries no indication of the affected MEP(s) with which the sender had detected a LoC defect.

3.1.2 On-Demand Fault Management Functions

On-demand fault management functions are initiated manually by the network operator and continue for a time bound period. These functions enable the operator to run diagnostics to investigate a defect condition.

3.1.2.1 Connectivity Verification

EVPN Network OAM MUST support on-demand connectivity verification mechanisms for unicast and multicast destinations. The connectivity verification mechanisms SHOULD provide a means for specifying and carrying in the messages:

- variable length payload/padding to test MTU related connectivity problems.

- test frame formats as defined in Appendix C of [RFC2544] to detect potential packet corruption.

EVPN Network OAM MUST support connectivity verification at per flow

granularity. This includes both user flows (to test a specific path between PEs) as well as test flows (to rest a representative path between PEs).

EVPN Service OAM MUST support connectivity verification on test flows and MAY support connectivity verification on user flows.

For multicast connectivity verification, EVPN Network OAM MUST support reporting on:

- the DF filtering status of specific port(s) or all the ports in a given bridge-domain.

- the Split Horizon filtering status of specific port(s) or all the ports in a given bridge-domain.

3.1.2.2 Fault Isolation

EVPN OAM MUST support an on-demand fault localization function. This involves the capability to narrow down the locality of a fault to a particular port, link or node. The characteristic of forward/reverse path asymmetry, in MPLS/IP, renders fault isolation into a direction-sensitive operation. That is, given two PEs A and B, localization of continuity failures between them requires running fault isolation procedures from PE A to PE B as well as from PE B to PE A.

EVPN Service OAM mechanisms only have visibility to the PEs but not the MPLS/IP P nodes. As such, they can be used to deduce whether the fault is in the customer's own network, the local CE-PE segment or remote CE-PE segment(s). EVPN Network and Transport OAM mechanisms can be used for fault isolation between the PEs and P nodes.

3.2 Performance Management

Performance Management functions can be performed both proactively and on-demand. Proactive management involves a recurring function, where the performance management probes are run continuously without a trigger. We cover both proactive and on-demand functions in this section.

3.2.1 Packet Loss

EVPN Network OAM SHOULD provide mechanisms for measuring packet loss for a given service.

Given that EVPN provides inherent support for multipoint-to-multipoint connectivity, then packet loss cannot be accurately measured by means of counting user data packets. This is because user packets can be delivered to more PEs or more ports than are necessary (e.g. due to broadcast, un-pruned multicast or unknown unicast flooding). As such, a statistical means of approximating packet loss rate is required. This can be achieved by sending "synthetic" OAM packets that are counted only by those ports (MEPs) that are required to receive them. This provides a statistical approximation of the number of data frames lost, even with multipoint-to-multipoint connectivity.

3.2.2 Packet Delay

EVPN Service OAM SHOULD support measurement of one-way and two-way packet delay and delay variation (jitter) across the EVPN network. Measurement of one-way delay requires clock synchronization between the probe source and target devices. Mechanisms for clock synchronization are outside the scope of this document. Note that Service OAM performance management mechanisms defined in [Y.1731] can be used.

EVPN Network OAM MAY support measurement of one-way and two-way packet delay and delay variation (jitter) across the EVPN network.

4. Security Considerations

EVPN OAM must provide mechanisms for:

- Preventing denial of service attacks caused by exploitation of the OAM message channel.

- Optionally authenticate communicating endpoints (MEPs and MIPs)

- Preventing OAM packets from leaking outside of the EVPN network or outside their corresponding Maintenance Domain. This can be done by having MEPs implement a filtering function based on the Maintenance Level associated with received OAM packets.

5. Acknowledgements

The authors would like to thank the following for their review of this work and valuable comments:

Gregory Mirsky, Alexander Vainshtein

6. IANA Considerations

This document requires no IANA actions.

Normative References

[RFC792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <https://www.rfc-editor.org/info/rfc792>.

[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>.

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

[RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <https://www.rfc-editor.org/info/rfc5881>.

[RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <https://www.rfc-editor.org/info/rfc5883>.

[RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <https://www.rfc-editor.org/info/rfc5884>.<

[RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", BCP 161, RFC 6291, DOI 10.17487/RFC6291, June 2011, <https://www.rfc-editor.org/info/rfc6291>.

[RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", RFC 6425, DOI 10.17487/RFC6425, November 2011, <https://www.rfc-editor.org/info/rfc6425>.

[RFC6428] Allan, D., Ed., Swallow, G., Ed., and J. Drake, Ed., "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, DOI 10.17487/RFC6428, November 2011, <https://www.rfc-editor.org/info/rfc6428>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February

2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

[RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <https://www.rfc-editor.org/info/rfc8029>.

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

Informative References

[802.1Q] "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks", 2014.

[Y.1731] "ITU-T Recommendation Y.1731 (02/08) - OAM functions and mechanisms for Ethernet based networks", February 2008.

[RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, DOI 10.17487/RFC2544, March 1999, <https://www.rfc-editor.org/info/rfc2544>.

[RFC5085] Nadeau, T., Ed., and C. Pignataro, Ed., "Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires", RFC 5085, DOI 10.17487/RFC5085, December 2007, <https://www.rfc-editor.org/info/rfc5085>.

[RFC6136] Sajassi, A., Ed., and D. Mohan, Ed., "Layer 2 Virtual Private Network (L2VPN) Operations, Administration, and Maintenance (OAM) Requirements and Framework", RFC 6136, DOI 10.17487/RFC6136, March 2011, <https://www.rfc-editor.org/info/rfc6136>.

Authors' Addresses

Samer Salam
Cisco

Email: ssalam@cisco.com

Ali Sajassi
Cisco
170 West Tasman Drive
San Jose, CA 95134, USA

Email: sajassi@cisco.com

Sam Aldrin
Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA USA

Email: aldrin.ietf@gmail.com

John E. Drake
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089, USA

Email: jdrake@juniper.net

Donald E. Eastlake, 3rd
Huawei Technologies
1424 Pro Shop Court
Davenport, FL 33896 USA

Tel: +1-508-333-2270
Email: d3e3e3@gmail.com

BESS Workgroup
Internet Draft

Intended status: Standards Track

J. Rabadan, Ed.
J. Kotalwar
S. Sathappan
Nokia

E. Rosen
Z. Zhang
W. Lin
Juniper

Expires: April 25, 2019

October 22, 2018

Multicast Source Redundancy in EVPN Networks
draft-skr-bess-evpn-redundant-mcast-source-00

Abstract

EVPN supports intra and inter-subnet IP multicast forwarding. However, EVPN (or conventional IP multicast techniques for that matter) do not have a solution for the case where a given multicast group carries more than one flow (i.e., more than one source), but where it is desired that each receiver gets only one of the several flows. Existing multicast techniques assume there are no redundant sources sending the same flows to the same IP multicast group, and, in case there were redundant sources, the receiver's application would deal with the received duplicated packets. This document extends the existing EVPN specifications and assumes that IP Multicast source redundancy may exist. It also assumes that, in case two or more sources send the same IP Multicast flows into the tenant domain, the EVPN PEs need to avoid that the receivers get packet duplication by following the described procedures.

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

This Internet-Draft will expire on April 25, 2019.

Table of Contents

1. Introduction

Intra and Inter-subnet IP Multicast forwarding are supported in EVPN networks. [IGMP-PROXY] describes the procedures required to optimize the delivery of IP Multicast flows when Sources and Receivers are connected to the same EVPN BD (Broadcast Domain), whereas [OISM] specifies the procedures to support Inter-subnet IP Multicast in a tenant network. Inter-subnet IP Multicast means that IP Multicast Source and Receivers of the same multicast flow are connected to different BDs of the same tenant.

[IGMP-PROXY], [OISM] or conventional IP multicast techniques do not have a solution for the case where a given multicast group carries more than one flow (i.e., more than one source), but where it is desired that each receiver gets only one of the several flows. Multicast techniques assume there are no redundant sources sending the same flows to the same IP multicast group, and, in case there were redundant sources, the receiver's application would deal with the received duplicated packets.

As a workaround in conventional IP multicast (PIM or MVPN networks), if all the redundant sources are given the same IP address, each receiver will get only one flow. The reason is that, in conventional IP multicast, (S,G) state is always created. It is always created by the RP, and sometimes by the Last Hop Router (LHR). The (S,G) state always binds the (S,G) flow to a source-specific tree, rooted at the source IP address. If multiple sources have the same IP address, one may end up with multiple (S,G) trees. However, the way the trees are constructed ensures that any given LHR or RP is on at most one of them. The use of an anycast address assigned to multiple sources may be useful for warm standby redundancy solutions. However, on one hand, it's not really helpful for hot standby redundancy solutions and on the other hand, configuring the same IP address (in particular IPv4 address) in multiple sources may bring issues if the sources need to be reached by IP unicast traffic or if the sources are attached to the same Broadcast Domain.

In addition, in the scenario where several G-sources are attached via EVPN/OISM, there is not necessarily any (S,G) state created for the redundant sources. In general, the LHRs have only (*,G) state, and there may not be an RP (creating (S,G) state) either. Therefore, this document extends the above two specifications and assumes that IP Multicast source redundancy may exist. It also assumes that, in case two or more sources send the same IP Multicast flows into the tenant domain, the EVPN PEs need to avoid that the receivers get packet duplication.

The solution provides support for Warm Standby (WS) and Hot Standby (HS) redundancy. WS is defined as the redundancy scenario in which the upstream PEs attached to the redundant sources of the same tenant, make sure that only one source of the same flow can send multicast to the interested downstream PEs at the same time. In HS the upstream PEs forward the redundant multicast flows to the downstream PEs, and the downstream PEs make sure only one flow is forwarded to the interested attached receivers.

1.1 Terminology

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

- OISM: Optimized Inter-Subnet Multicast, as in [OISM].
- Broadcast Domain (BD): an emulated ethernet, such that two systems on the same BD will receive each other's link-local broadcasts. In this document, BD also refers to the instantiation of a Broadcast Domain on an EVPN PE. An EVPN PE can be attached to one or multiple BDs of the same tenant.
- Designated Forwarder (DF): as defined in [RFC7432], an ethernet segment may be multi-homed (attached to more than one PE). An ethernet segment may also contain multiple BDs, of one or more EVIs. For each such EVI, one of the PEs attached to the segment becomes that EVI's DF for that segment. Since a BD may belong to only one EVI, we can speak unambiguously of the BD's DF for a given segment.
- Upstream PE: in this document an Upstream PE is referred to as the EVPN PE that is connected to the IP Multicast source or closest to it. It receives the IP Multicast flows on local ACs (Attachment Circuits).

- Downstream PE: in this document a Downstream PE is referred to as the EVPN PE that is connected to the IP Multicast receivers and gets the IP Multicast flows from remote EVPN PEs.

- G-traffic: any frame with an IP payload whose IP Destination Address (IP DA) is a multicast group G.

- G-source: any system sourcing traffic to G.

- SFG: Single Flow Group, i.e., a multicast group address G which represents traffic that contains only a single flow. However, multiple sources - with the same or different IP - may be transmitting an SFG.

- Redundant G-source: a host or router that transmits an SFG in a tenant network where there are more hosts or routers transmitting the same SFG. Redundant G-sources for the same SFG SHOULD have different IP addresses when in the same BD, and MAY have the same IP address when in different BDs of the same tenant network. Redundant G-sources are assumed NOT to be "bursty" in this document (typical example are Broadcast TV G-sources or similar).

- P-tunnel: Provider tunnel refers to the type of tree a given upstream EVPN PE uses to forward multicast traffic to downstream PEs. Examples of P-tunnels supported in this document are Ingress Replication (IR), Assisted Replication (AR), BIER, mLDP or P2MP RSVP-TE.

- Inclusive Multicast Tree or Inclusive Provider Multicast Service Interface (I-PMSI): defined in [RFC6513], in this document it is applicable only to EVPN and refers to the default multicast tree for a given BD. All the EVPN PEs that are attached to a specific BD belong to the I-PMSI for the BD. The I-PMSI trees are signaled by EVPN Inclusive Multicast Ethernet Tag (IMET) routes.

- Selective Multicast Tree or Selective Provider Multicast Service Interface (S-PMSI): defined in [RFC6513], in this document it is applicable only to EVPN and refers to the multicast tree to which only the interested PEs of a given BD belong to. There are two types of EVPN S-PMSIs:

  - EVPN S-PMSIs that require the advertisement of S-PMSI AD routes from the upstream PE, as in [EVPN-BUM]. The interested downstream PEs join the S-PMSI tree as in [EVPN-BUM].

  - EVPN S-PMSIs that don’t require the advertisement of S-PMSI AD routes. They use the forwarding information of the IMET routes, but upstream PEs send IP Multicast flows only to downstream PEs

issuing Selective Multicast Ethernet Tag (SMET) routes for the flow. These S-PMSIs are only supported with the following P-tunnels: Ingress Replication (IR), Assisted Replication (AR) and BIER.

This document also assumes familiarity with the terminology of [RFC7432], [RFC4364], [RFC6513], [RFC6514], [IGMP-PROXY], [OISM], [EVPN-RT5] and [EVPN-BUM].

1.2 Background on IP Multicast Delivery in EVPN Networks

IP Multicast is all about forwarding a single copy of a packet from a source S to a group of receivers G along a multicast tree. That multicast tree can be created in an EVPN tenant domain where S and the receivers for G are connected to the same BD or different BD. In the former case, we refer to Intra-subnet IP Multicast forwarding, whereas the latter case will be referred to as Inter-subnet IP Multicast forwarding.

1.2.1 Intra-subnet IP Multicast Forwarding

When the source S1 and receivers interested in G1 are attached to the same BD, the EVPN network can deliver the IP Multicast traffic to the receivers in two different ways (Figure 1):

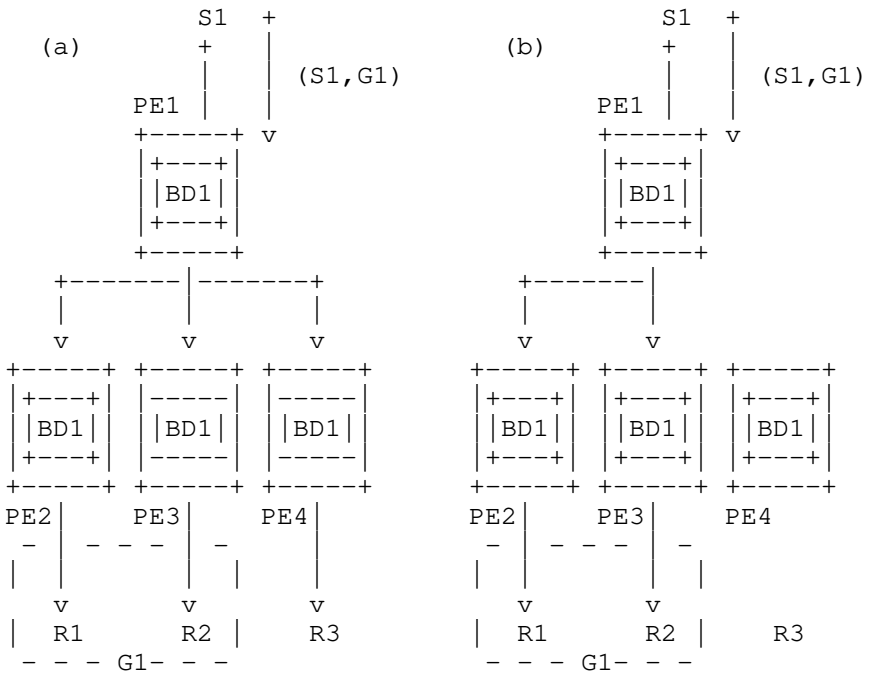


Figure 1 - Intra-subnet IP Multicast

Model (a) illustrated in Figure 1 is referred to as IP Multicast delivery as BUM traffic. This way of delivering IP Multicast traffic does not require any extensions to [RFC7432], however, it sends the IP Multicast flows to non-interested receivers, such as e.g., R3 in Figure 1. In this example, downstream PEs can snoop IGMP/MLD messages from the receivers so that layer-2 multicast state is created and, for instance, PE4 can avoid sending (S1,G1) to R3, since R3 is not interested in (S1,G1).

Model (b) in Figure 1 uses an S-PMSI to optimize the delivery of the (S1,G1) flow. For instance, assuming PE1 uses IR, PE1 sends (S1,G1) only to the downstream PEs that issued an SMET route for (S1,G1), that is, PE2 and PE3. In case PE1 uses any P-tunnel different than IR, AR or BIER, PE1 will advertise an S-PMSI A-D route for (S1,G1) and PE2/PE2 will join that tree.

Procedures for Model (b) are specified in [IGMP-PROXY].

1.2.2 Inter-subnet IP Multicast Forwarding

If the source and receivers are attached to different BDs of the same

tenant domain, the EVPN network can also use Inclusive or Selective Trees as depicted in Figure 2, models (a) and (b) respectively.

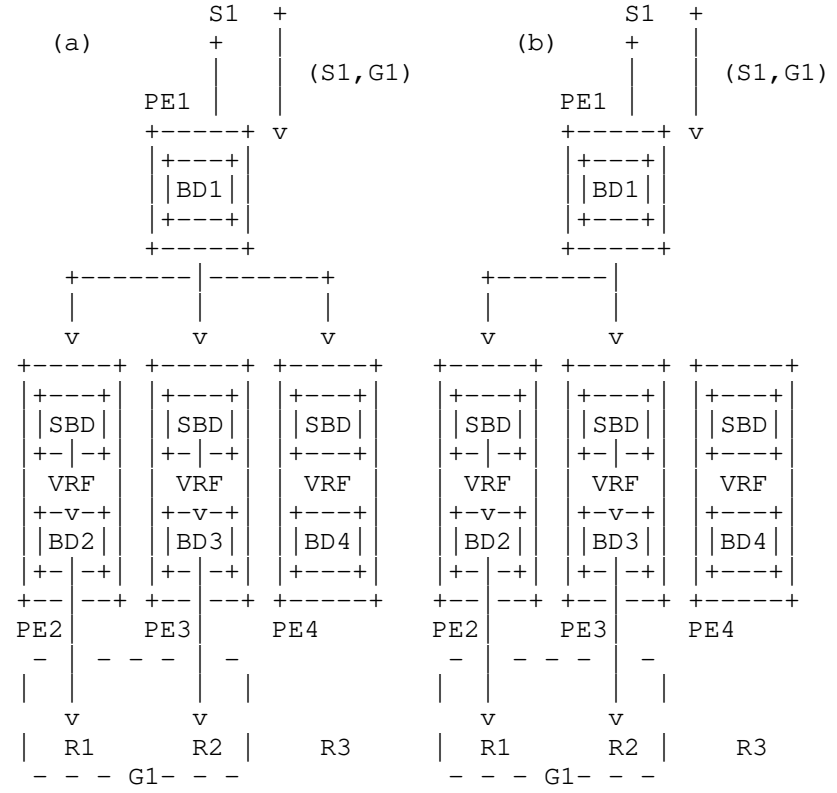


Figure 2 - Inter-subnet IP Multicast

[OISM] specifies the procedures to optimize the Inter-subnet Multicast forwarding in an EVPN network. The IP Multicast flows are always sent in the context of the source BD. As described in [OISM], if the downstream PE is not attached to the source BD, the IP Multicast flow is received on the SBD (Supplementary Broadcast Domain), as in the example in Figure 2.

[OISM] supports Inclusive or Selective Multicast Trees, and as explained in section 1.3.1 "Intra-subnet IP Multicast Forwarding", the Selective Multicast Trees are setup in a different way, depending on the P-tunnel being used by the source BD. As an example, model (a) in Figure 2 illustrates the use of an Inclusive Multicast Tree for

BD1 on PE1. Since the downstream PEs are not attached to BD1, they will all receive (S1,G1) in the context of the SBD and will locally route the flow to the local ACs. Model (b) uses a similar forwarding model, however PE1 sends the (S1,G1) flow in a Selective Multicast Tree. If the P-tunnel is IR, AR or BIER, PE1 does not need to advertise an S-PMSI A-D route.

[OISM] is a superset of the procedures in [IGMP-PROXY], in which sources and receivers can be in the same or different BD of the same tenant. [OISM] ensures every upstream PE attached to a source will learn of all other PEs (attached to the same Tenant Domain) that have interest in a particular set of flows. This is because the downstream PEs advertise SMET routes for a set of flows with the SBD's Route Target and they are imported by all the Upstream PEs of the tenant. As a result of that, inter-subnet multicasting can be done within the Tenant Domain, without requiring any Rendezvous Points (RP), shared trees, UMH selection or any other complex aspects of conventional multicast routing techniques.

1.3 Multi-Homed IP Multicast Sources in EVPN

Contrary to conventional multicast routing technologies, multi-homing PEs attached to the same source can never create IP Multicast packet duplication if the PEs use a multi-homed Ethernet Segment (ES). Figure 3 illustrates this by showing two multi-homing PEs (PE1 and PE2) that are attached to the same source (S1). We assume that S1 is connected to an all-active ES by a layer-2 switch (SW1) with a LAG to PE1 and PE2.

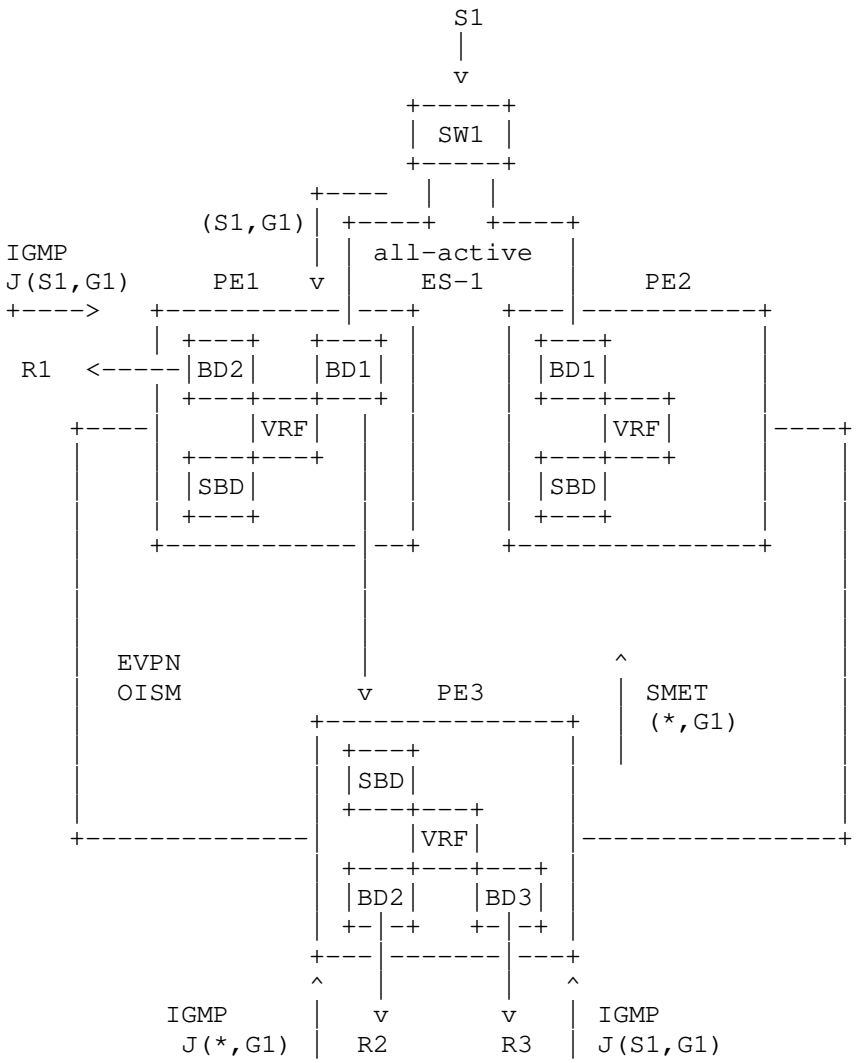


Figure 3 - All-active Multi-homing and OISM

When receiving the (S1,G1) flow from S1, SW1 will choose only one link to send the flow, as per [RFC7432]. Assuming PE1 is the receiving PE on BD1, the IP Multicast flow will be forwarded as soon as BD1 creates multicast state for (S1,G1) or (*,G1). In the example of Figure 3, receivers R1, R2 and R3 are interested in the multicast flow to G1. R1 will receive (S1,G1) directly via the IRB interface as per [OISM]. Upon receiving IGMP reports from R2 and R3, PE3 will issue an SMET (*,G1) route that will create state in PE1's BD1. PE1

will therefore forward the IP Multicast flow to PE3's SBD and PE3 will forward to R2 and R3, as per [OISM] procedures.

When IP Multicast source multi-homing is required, EVPN multi-homed Ethernet Segments SHOULD be used. EVPN multi-homing guarantees that only one Upstream PE will forward a given multicast flow at the time, avoiding packet duplication at the Downstream PEs. In addition, the SMET route for a given flow creates state in all the multi-homing Upstream PEs. Therefore, in case of failure on the Upstream PE forwarding the flow, the backup Upstream PE can forward the flow immediately.

This document assumes that multi-homing PEs attached to the same source always use multi-homed Ethernet Segments.

1.4 The Need for Redundant IP Multicast Sources in EVPN

While multi-homing PEs to the same IP Multicast G-source provides certain level of resiliency, multicast applications are often critical in the Operator's network and greater level of redundancy is required. This document assumes that:

a) Redundant G-sources for an SFG may exist in the EVPN tenant network. A Redundant G-source is a host or a router that sends an SFG in a tenant network where there is another host or router sending traffic to the same SFG.

b) Those redundant G-sources may be in the same BD or different BDs of the tenant. There must not be restrictions imposed on the location of the receiver systems either.

c) The redundant G-sources can be single-homed to only one EVPN PE or multi-homed to multiple EVPN PEs.

d) The EVPN PEs must avoid duplication of the same SFG on the receiver systems.

2. Solution Overview

There are two redundant G-source solutions described in this document:

- Warm Standby (WS) Solution
- Hot Standby (HS) Solution

The WS solution is an upstream PE based solution (downstream PEs do not participate in the procedures), in which all the upstream PEs attached to redundant G-sources for an SFG will elect a "Single Forwarder" (SF) among themselves. Once a SF is elected, the upstream PEs add an RPF check to the (*,G) state for the SFG:

- A non-SF upstream PE discards any (*,G) packets received over a local AC.

- The SF accepts and forwards any (*,G) packets it receives over a single local AC. In case (*,G) packets are received over multiple local ACs, they will be discarded in all the local ACs but one. The procedure to choose the local AC that accepts packets is a local implementation matter.

A failure on the SF will result in the election of a new SF. The Election requires BGP extensions on the existing EVPN routes. These extensions and associated procedures are described in Sections 3 and 4 respectively.

In the HS solution the downstream PEs are the ones avoiding the SFG duplication. The upstream PEs are aware of the locally attached G-sources and add a unique ESI-label per SFG to the SFG packets forwarded to downstream PEs. The downstream PEs pull the SFG from all the upstream PEs attached to the redundant G-sources and avoid duplication on the receiver systems by adding an RPF check to the (*,G) state for the SFG:

- A downstream PE discards any (*,G) packets it receives from the "wrong G-source".

- The wrong G-source is identified in the data path by an ESI-label that is different than the ESI-label used for the selected G-source.

- Note that the ESI-label is used here for "ingress filtering" as opposed to the [RFC7432] "egress filtering" used in the split-horizon procedures. In [RFC7432] the ESI-label indicates what egress ACs must be skipped when forwarding BUM traffic to the egress. In this document, the ESI-label indicates what ingress traffic must be discarded.

The use of ESI-labels for SFGs forwarded by upstream PEs require some control plane and data plane extensions in the procedures used by [RFC7432] for multi-homing. Upon failure of the selected G-source, the downstream PE will switch over to a different selected G-source, and will therefore change the RPF check for the (*,G) state. The extensions and associated procedures are described in Sections 3 and

5 respectively.

An operator should use the HS solution if they require a fast fail-over time and the additional bandwidth consumption is acceptable (SFG packets are received multiple times on the downstream PEs). Otherwise the operator should use the WS solution, at the expense of a slower fail-over time in case of a G-source or upstream PE failure. Besides bandwidth efficiency, another advantage of the WS solution is that only the upstream PEs attached to the redundant G-sources for the same SFG need to be upgraded to support the new procedures.

The support of either solution is OPTIONAL.

3. BGP EVPN Extensions

This document makes use of the following BGP EVPN extensions:

1. SFG flag in the Multicast Flags Extended Community

   The Single Flow Group (SFG) flag is a new bit requested to IANA out of the registry Multicast Flags Extended Community Flag Values. This new flag is set for S-PMSI routes that carry an SFG (*,G) in the NLRI.

2. ESI attribute

   The HS solution requires the advertisement of one or more attributes that encode the Ethernet Segment Identifier(s) associated to an S-PMSI (*,G) route that advertises the presence of an SFG. The format of this attribute will be described in future revisions of this document. The following options are being considered for the "ESI attribute":

   - Use a BGP Large Community (LC) Attribute:

     If an Ethernet Segment Type 5 [RFC7432] is used for ESes attached to redundant G-sources, a LC attribute can be used where each value encodes the corresponding ESI in the following format: ASN(4-bytes):Local-Discriminator(4-bytes):0x0(4-bytes); ASN and Local-Discriminator are the same values that are used at the upstream PE to construct the type-5 ESI. A PE receiving an S-PMSI (*,G) route with an SFG indication should interpret the LC Attribute as a list of ESIs associated with the redundant G-sources.

   - Use a new BGP attribute

Another option is to define a new attribute that can encode one or more ESI values.

- Use an IPv6 Address Specific BGP Extended Community Attribute

  Another Option is to make use of the [RFC5701] IPv6 Address EC attribute.

This section will be modified in future versions of the document.

4. Warm Standby (WS) Solution for Redundant G-Sources

The general procedure is described as follows:

1. Configuration of the upstream PEs

   Upstream PEs where redundant G-sources may exist need to be configured to know which groups are carrying only flows from redundant G-sources, that is, the SFGs in the tenant domain. They will also be configured to know which local BDs may be attached to a redundant G-source. As an example, PE1 is configured to know that G1 is an SFG and redundant G-sources for G1 may be attached to BD1 or BD2.

2. Signaling the location of a G-source for a given SFG

   Upon receiving G-traffic for an SFG on a BD, an upstream PE configured to follow this procedure, e.g., PE1:

   a. Originates an S-PMSI (*,G) route for the SFG that is imported by all the PEs attached to the tenant domain. In order to do that, the route will use the SBD-RT (Supplementary Broadcast Domain Route-Target) in addition to the BD-RT of the BD over which the G-traffic is received. The route SHOULD also carry a DF Election Extended Community (EC) and a flag indicating that it conveys an SFG. The DF Election EC and its use is specified in [DF].

   b. The above S-PMSI route MAY be advertised with or without PMSI Tunnel Attribute (PTA):

      - With no PTA if an I-PMSI or S-PMSI with IR/AR/BIER are to be used.

      - With PTA in any other case.

   c. The S-PMSI (*,G) route is triggered by the first packet of the

SFG and withdrawn when the flow is not received anymore. Detecting when the G-source is no longer active is a local implementation matter. The use of a timer is RECOMMENDED. The timer is started when the traffic to G1 is not received. Upon expiration of the timer, the PE will withdraw the route.

3. Single Forwarder (SF) Election

   If the PE with a local G-source receives an S-PMSI route for the same SFG from a remote PE, it will run a Single Forwarder (SF) Election based on the information encoded in the DF Election EC.

4. RPF check on the PEs attached to a redundant G-source

   All the PEs with a local G-source for the SFG will add an RPF check to the (*,G) state for the SFG. That RPF check depends on the SF Election result:

   a. The non-SF PEs discard any (*,G) packets received over a local AC.

   b. The SF accepts any (*,G) packets it receives over one (and only one) local AC.

The solution above provides redundancy for SFGs and it does not require an upgrade of the downstream PEs (PEs where there is certainty that no redundant G-sources are connected). Other G-sources for non-SFGs may exist in the same tenant domain. This document does not change the existing procedures for non-SFG G-sources.

The redundant G-sources can be single-homed or multi-homed to a BD in the tenant domain. Multi-homing does not change the above procedures.

Sections 4.1 and 4.2 show two examples of the WS solution.

4.1 WS Example in an OISM Network

Figure 4 illustrates an example in which S1 and S2 are redundant G-sources for the SFG (*,G1).

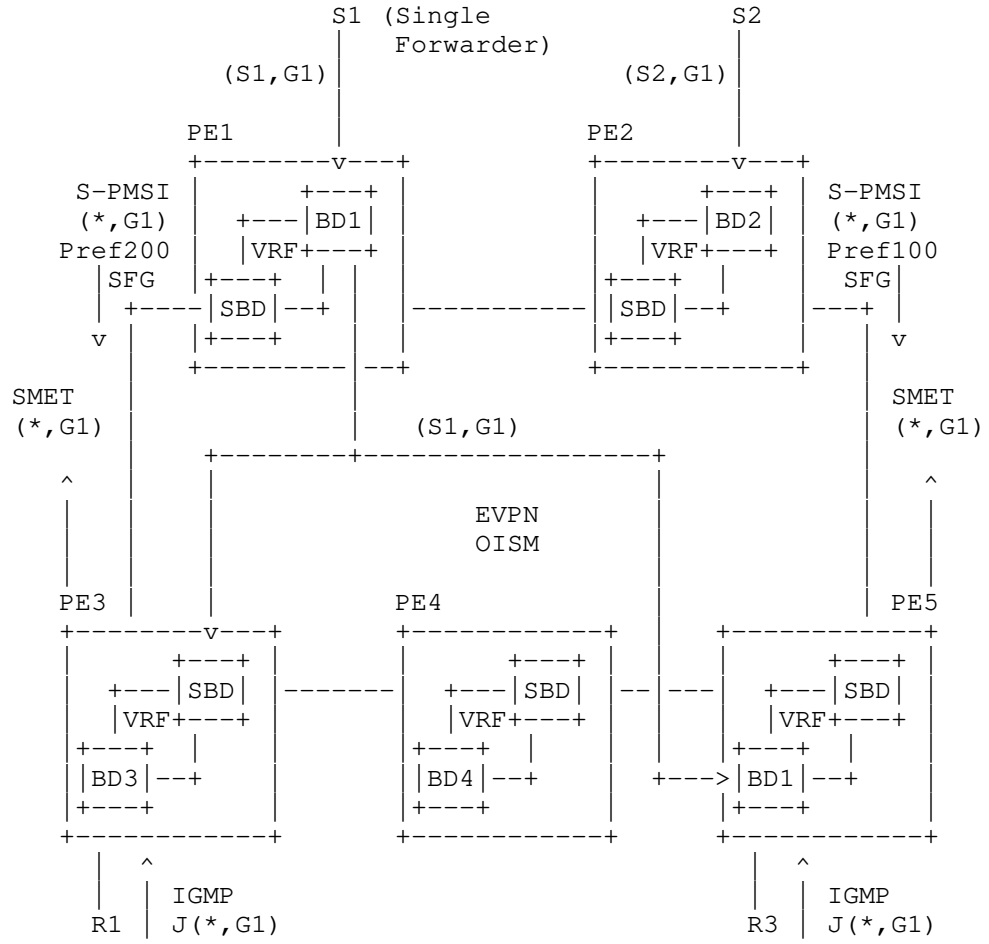


Figure 4 - WS Solution for Redundant G-Sources

The WS solution works as follows:

1. Configuration of the upstream PEs, PE1 and PE2

   PE1 and PE2 are configured to know that G1 is an SFG and redundant G-sources for G1 may be attached to BD1 or BD2, respectively.

2. Signaling the location of S1 and S2 for (*,G1)

   Upon receiving (S1,G1) traffic on a local AC, PE1 and PE2 originate S-PMSI (*,G1) routes with the SBD-RT, DF Election Extended Community (EC) and a flag indicating that it conveys an

SFG.

3. Single Forwarder (SF) Election

   Based on the DF Election EC content, PE1 and PE2 elect an SF for (*,G1). Assuming both PEs agree on e.g., Preference based Election as the algorithm to use [DF-PREF], and PE1 has a higher preference, PE1 becomes the SF for (*,G1).

4. RPF check on the PEs attached to a redundant G-source

   a. The non-SF, PE2, discards any (*,G1) packets received over a local AC.

   b. The SF, PE1 accepts (*,G1) packets it receives over a one (and only one) local AC.

The end result is that, upon receiving reports for (*,G1) or (S,G1), the downstream PEs (PE3 and PE5) will issue SMET routes and will pull the multicast SFG from PE1, and PE1 only. A failure on S1, the AC connected to S1 or PE1 itself will trigger the S-PMSI (*,G1) withdrawal from PE1 and PE2 will be promoted to SF.

4.2 WS Example in a Single-BD Tenant Network

Figure 5 illustrates an example in which S1 and S2 are redundant G-sources for the SFG (*,G1), however, now all the G-sources and receivers are connected to the same BD1 and there is no SBD.

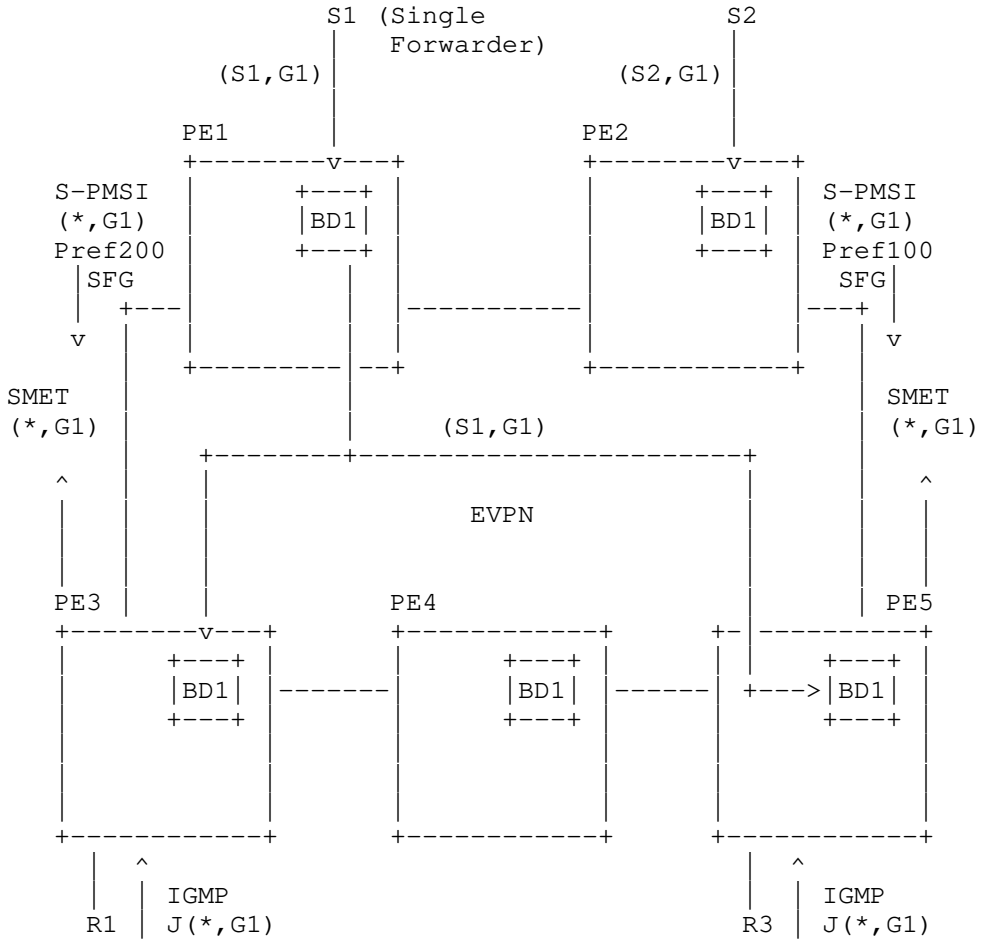


Figure 5 - WS Solution for Redundant G-Sources in the same BD

The same procedure as in Section 4.1 is valid here, being this a sub-case of the one in Section 4.1. Upon receiving traffic for the SFG G1, PE1 and PE2 advertise the S-PMSI routes with BD1-RT only, since there is no SBD.

5. Hot Standby (HS) Solution for Redundant G-Sources

If fast-failover time is desired upon the failure of a G-source or PE attached to the G-source, and in spite of the extra bandwidth consumption in the tenant network, the HS solution should be used. The procedure is as follows:

1. Configuration of the PEs

   As in the WS case, the upstream PEs where redundant G-sources may exist need to be configured to know which groups are carrying only flows from redundant G-sources, that is, the SFGs in the tenant domain. In addition (and this is not done in WS) the individual redundant G-sources for an SFG need to be associated with an Ethernet Segment (ES) on the upstream PEs:

   - This is irrespective of the redundant G-source being multi-homed or single-homed. Even for single-homed redundant G-sources the HS procedure relies on the ESI labels for the RPF check on downstream PEs. The term "S-ESI" is used in this document to refer to an ESI associated to a redundant G-source.

   - The S-ESI SHOULD be a Type 5 ESI [RFC7432] so that it can be mapped to a value in a BGP LC attribute, as described in Section 3. The S-ESI MAY also be configured.

   Contrary to the WS method (that is transparent to the downstream PEs), the support for the HS procedure in all downstream PEs connected to the receivers in the tenant network is REQUIRED. The downstream PEs do not need to be configured to know the connected SFGs or their ESIs, since they get that information from the upstream PEs. The downstream PEs will locally select an ESI for a given SFG, and will program an RPF check to the (*,G) state for the SFG that will discard (*,G) packets from the rest of the ESIs. The selection of the ESI for the SFG is based on local policy.

2. Signaling the location of a G-source for a given SFG and its association to the local ESIs

   Based on the configuration in step 1, an upstream PE configured to follow the HS procedures:

   a. Advertises an S-PMSI (*,G) route per each configured SFG. These routes need to be imported by all the PEs of the tenant domain, therefore they will carry the BD-RT and SBD-RT (if the SBD exists). The route also carries the ESI attribute that conveys all the S-ESIs associated to the SFG in the PE.

   b. The S-PMSI route will convey a PTA if the same cases as in the WS procedure.

   c. The S-PMSI (*,G) route is triggered by the configuration of the SFG and not by the reception of G-traffic.

3. Distribution of DCB (Domain-wide Common Block) ESI-labels and G-

source ES routes

An upstream PE advertises the corresponding ES, A-D per-EVI and A-D per-ES routes for the local S-ESIs.

a. ES routes are used for regular DF Election for the S-ES. This document does not introduce any change in the procedures related to the ES routes.

b. The A-D per-EVI and A-D per-ES routes MUST include the SBD-RT since they have to be imported by all the PEs in the tenant domain.

c. The A-D per-ES routes convey the S-ESI labels that the downstream PEs use to add the RPF check for the (*,G) associated to the SFGs. This RPF check requires that all the packets for a given G-source are received with the same S-ESI label value on the downstream PEs. For example, if two redundant G-sources are multi-homed to PE1 and PE2 via S-ES-1 and S-ES-2, PE1 and PE2 MUST allocate the same ESI label "Lx" for S-ES-1 and they MUST allocate the same ESI label "Ly" for S-ES-2. In addition, Lx and Ly MUST be different. These ESI labels are Domain-wide Common Block (DCB) labels and follow the procedures in [DCB].

4. Processing of A-D routes and RPF check on the downstream PEs

Unless described otherwise, "A-D routes" in this section refers to both types, A-D per-ES and A-D per-EVI routes. The A-D routes are received and imported in all the PEs in the tenant domain. The processing of the A-D routes on a given PE depends on its configuration:

a. The PEs attached to the same BD of the BD-RT that is included in the A-D routes will process the routes as in [RFC7432] and [DF]. If the receiving PE is attached to the same ES as indicated in the route, [RFC7432] split-horizon procedures will be followed and the DF Election candidate list may be modified as in [DF] if the ES supports the AC-DF capability.

b. The PEs that are not attached to the BD-RT but are attached to the SBD of the received SBD-RT, will import the A-D routes and use them for redundant G-source mass withdrawal, as explained later.

c. Upon importing A-D per-ES routes corresponding to different S-ESes, a PE MUST select a primary S-ES and add an RPF check to the (*,G) state in the BD or SBD. This RPF check will discard

all ingress packets to (*,G) that are not received with the ESI-label of the primary S-ES. The selection of the primary S-ES is a matter of local policy.

5. G-traffic forwarding for redundant G-sources and fault detection

Assuming there is (*,G) or (S,G) state for the SFG with OIF list entries associated to remote EVPN PEs, upon receiving G-traffic on a S-ES, the upstream PE will add a S-ESI label at the bottom of the stack before forwarding the traffic to the remote EVPN PEs. This label is allocated from a DCB as described in step 3. If P2MP or BIER PMSIs are used, this is not adding any new data path procedures on the upstream PEs (except that the ESI-label is allocated from a DCB). However, if IR/AR are used, this document extends the [RFC7432] procedures by pushing the S-ESI labels not only on packets sent to the PEs that shared the ES but also to the rest of the PEs in the tenant domain. This allows the downstream PEs to receive all the multicast packets from the redundant G-sources with a S-ESI label (irrespective of the PMSI type and the local ESes), and discard any packet that conveys a S-ESI label different from the primary S-ESI label (that is, the label associated to the selected primary S-ES), as discussed in step 4.

If the last A-D per-EVI or the last A-D per-ES route for the primary S-ES is withdrawn, the downstream PE will immediately select a new primary S-ES and will change the RPF check. Note that if the S-ES is re-used for multiple tenant domains by the upstream PEs, the withdrawal of all the A-D per-ES routes for a S-ES provides a mass withdrawal capability that makes a downstream PE to change the RPF check in all the tenant domains using the same S-ES.

The withdrawal of the last S-PMSI route for a given (*,G) SHOULD make the downstream PE remove the S-ESI label based RPF check on (*,G).

5.1 Use of BFD in the HS Solution

This section will be completed in a future version of this document.

5.2 HS Example in an OISM Network

Figure 6 illustrates the HS model in an OISM network. As in previous examples, S1 and S2 are redundant G-sources for the SFG (*,G1) in BD1. S1 and S2 are (all-active) multi-homed to upstream PEs, PE1 and PE2. The receivers are attached to downstream PEs, PE3 and PE5, in BD3 and BD1, respectively. S1 and S2 are assumed to be connected by a

LAG to the multi-homing PEs, and the multicast traffic can use the link to either upstream PE. The diagram illustrates how S1 sends the G-traffic to PE1 and PE1 forwards to the remote interested downstream PEs, whereas S2 sends to PE2 and PE2 forwards further. In this HS model, the interested downstream PEs will get duplicate G-traffic from the two G-sources for the same SFG. While the diagram shows that the two flows are forwarded by different upstream PEs, the all-active multi-homing procedures may cause that the two flows come from the same upstream PE. Therefore, finding out the upstream PE for the flow is not enough for the downstream PEs to program the required RPF check to avoid duplicate packets on the receiver.

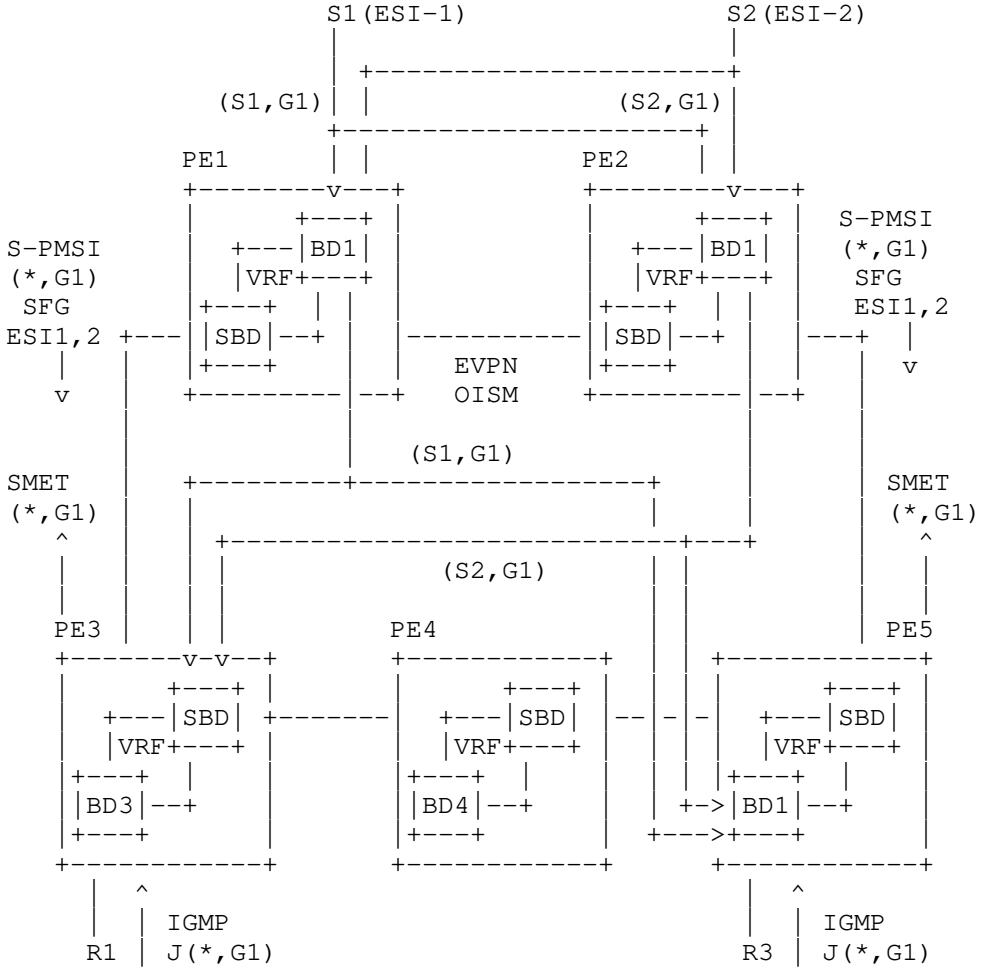


Figure 6 - HS Solution for Multi-homed Redundant G-Sources in OISM

In this scenario, the HS solution works as follows:

1. Configuration of the upstream PEs, PE1 and PE2

   PE1 and PE2 are configured to know that G1 is an SFG and the redundant G-sources for G1 use S-ESIs ESI-1 and ESI-2 respectively. Both ESes are configured in both PEs and the ESI value can be configured or auto-derived as an ES type 5. The ESI-label values are allocated from a DCB [DCB] and are configured either locally or by a centralized controller. We assume ESI-1 is configured to use ESI-label-1 and ESI-2 to use ESI-label-2.

The downstream PEs, PE3, PE4 and PE5 are configured to support HS mode and select the G-source with e.g., lowest ESI value.

2. PE1 and PE2 advertise S-PMSI (*,G1) and ES/A-D routes

   Based on the configuration of step 1, PE1 and PE2 advertise an S-PMSI (*,G1) route each. The route from each of the two PEs will include the ESI attribute with ESI-1 and ESI-2, as well as BD1-RT plus SBD-RT and a flag that indicates that G1 is an SFG.

   In addition, PE1 and PE2 advertise ES and A-D routes for ESI-1 and ESI-2. The A-D per-ES and per-EVI routes will include the SBD-RT so that they can be imported by the downstream PEs that are not attached to BD1, e.g., PE3 and PE4. The A-D per-ES routes will convey ESI-label-1 for ESI-1 (on both PEs) and ESI-label-2 for ESI-2 (also on both PEs).

3. Processing of A-D routes and RPF check

   PE1 and PE2 received each other's ES and A-D routes. Regular [RFC7432] [DF] procedures will be followed for DF Election and programming of the ESI-labels for egress split-horizon filtering. PE3/PE4 import the A-D routes in the SBD. Since PE3 has created a (*,G1) state based on local interest, PE3 will add an RPF check to (*,G1) so that packets coming with ESI-label-2 are discarded (lowest ESI value is assumed to give the primary S-ES).

4. G-traffic forwarding and fault detection

   PE1 receives G-traffic (S1,G1) on ES-1 that is forwarded within the context of BD1. Irrespective of the tunnel type, PE1 pushes ESI-label-1 at the bottom of the stack and the traffic gets to PE3 and PE5 with the mentioned ESI-label (PE4 has no local interested receivers). The G-traffic with ESI-label-1 passes the RPF check and it is forwarded to R1. In the same way, PE2 sends (S2,G1) with ESI-label-2, but this G-traffic does not pass the RPF check and gets discarded at PE3/PE5.

   If the link from S1 to PE1 fails, S1 will forward the (S1,G1) traffic to PE2 instead. PE1 withdraws the ES and A-D routes for ESI-1. Now both flows will be originated by PE2, however the RPF checks don't change in PE3/PE5.

   If subsequently, the link from S1 to PE2 fails, PE2 also withdraws the ES and A-D routes for ESI-1. Since PE3 and PE5 have no longer A-D routes for ESI-1, they immediately change the RPF check so that packets with ESI-label-2 are now accepted.

Figure 7 illustrates a scenario where S1 and S2 are single-homed to PE1 and PE2 respectively. This scenario is a sub-case of the one in Figure 6. Now ES-1 only exists in PE1, hence only PE1 advertises the A-D routes for ESI-1. Similarly, ES-2 only exists in PE2 and PE2 is the only PE advertising A-D routes for ESI-2. The same procedures as in Figure 6 applies to this use-case.

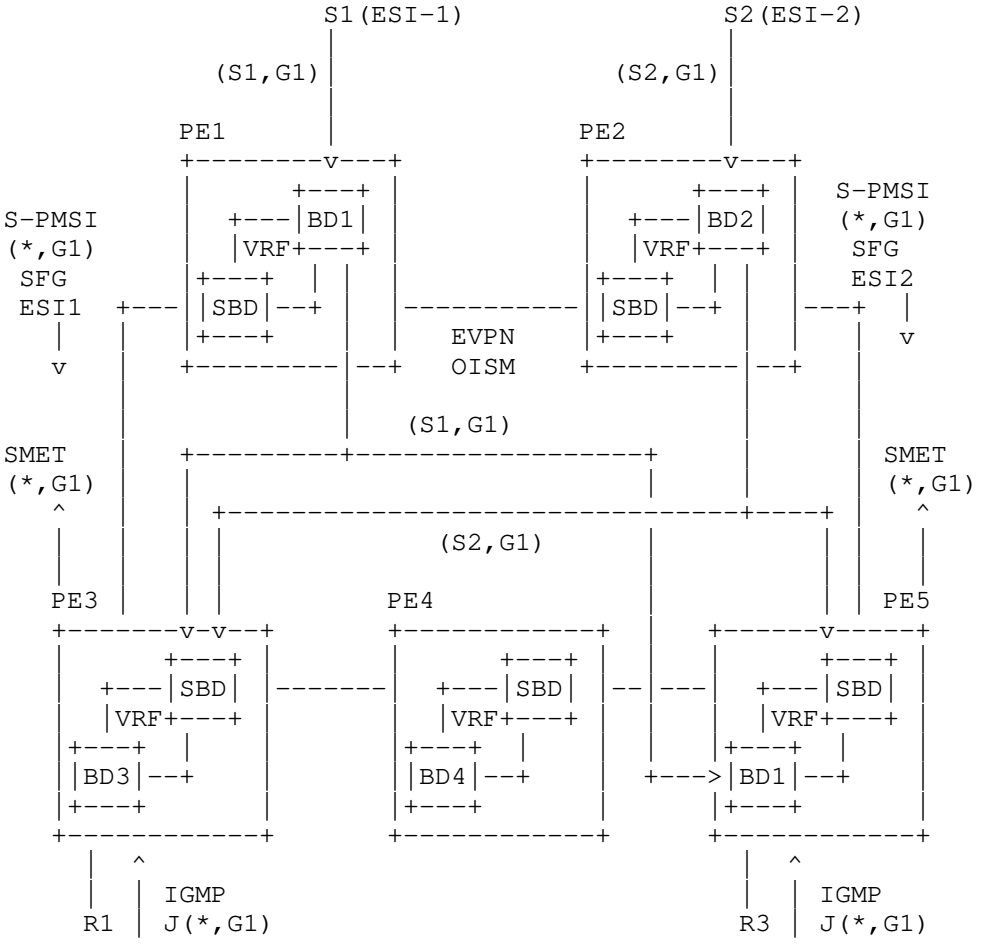


Figure 7 - HS Solution for single-homed Redundant G-Sources in OISM

5.3 HS Example in a Single-BD Tenant Network

Irrespective of the redundant G-sources being multi-homed or single-homed, if the tenant network has only one BD, e.g., BD1, the procedures of Section 5.2 still apply, only that routes do not include any SBD-RT and all the procedures apply to BD1 only.

6. Security Considerations

The same Security Considerations described in [OISM] are valid for this document.

7. IANA Considerations

IANA is requested to allocate a Bit in the Multicast Flags Extended Community.

8. References

8.1. Normative References

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC6513] Rosen, E., Ed., and R. Aggarwal, Ed., "Multicast in MPLS/BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <https://www.rfc-editor.org/info/rfc6513>.

[RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, DOI 10.17487/RFC6514, February 2012, <https://www.rfc-editor.org/info/rfc6514>.

[IGMP-PROXY] Sajassi, A. et al, "IGMP and MLD Proxy for EVPN", June 2018, work-in-progress, draft-ietf-bess-evpn-igmp-mld-proxy-02.

[OISM] Rosen, E. et al, "EVPN Optimized Inter-Subnet Multicast (OISM) Forwarding", June 2018, work-in-progress, draft-ietf-bess-evpn-irb-mcast-01.

[DF] Rabadan, J., Mohanty, S., Sajassi, A., Drake, J., Nagaraj, K., and S. Sathappan, "Framework for EVPN Designated Forwarder Election Extensibility", internet-draft draft-ietf-bess-evpn-df-election-framework-05.txt, 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>.

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

[DCB] Zhang, Z. et al, "MVPN/EVPN Tunnel Aggregation with Common Labels", April 2018, work-in-progress, draft-zzhang-bess-mvpn-evpn-aggregation-label-01.

8.2. Informative References

[EVPN-RT5] Rabadan, J., Henderickx, W., Drake, J., Lin, W., and A. Sajassi, "IP Prefix Advertisement in EVPN", internet-draft ietf-bess-evpn-prefix-advertisement-11.txt, May 2018.

[EVPN-BUM] Zhang, Z., Lin, W., Rabadan, J., and K. Patel, "Updates on EVPN BUM Procedures", internet-draft ietf-bess-evpn-bum-procedure-updates-03, April 2018.

[DF-PREF] Rabadan, J., Sathappan, S., Przygienda, T., Lin, W., Drake, J., Sajassi, A., and S. Mohanty, "Preference-based EVPN DF Election", internet-draft ietf-bess-evpn-pref-df-02.txt, October 2018.

[RFC5701] Rekhter, Y., "IPv6 Address Specific BGP Extended Community Attribute", RFC 5701, DOI 10.17487/RFC5701, November 2009, <https://www.rfc-editor.org/info/rfc5701>.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

9. Acknowledgments

10. Contributors

Authors' Addresses

Jorge Rabadan
Nokia
777 E. Middlefield Road
Mountain View, CA 94043 USA
Email: jorge.rabadan@nokia.com

Senthil Sathappan
Nokia
701 E. Middlefield Road
Mountain View, CA 94043 USA
Email: senthil.sathappan@nokia.com

Jayant Kotalwar
Nokia
701 E. Middlefield Road
Mountain View, CA 94043 USA
Email: jayant.kotalwar@nokia.com

Eric C. Rosen
Juniper Networks, Inc.
EMail: erosen@juniper.net

Zhaohui Zhang
Juniper Networks
EMail: zzhang@juniper.net

Wen Lin
Juniper Networks, Inc.
EMail: wlin@juniper.net

BESS Workgroup
Internet-Draft
Intended status: Standards Track
Expires: May 6, 2021

J. Rabadan, Ed.
J. Kotalwar
S. Sathappan
Nokia
Z. Zhang
W. Lin
Juniper
E. Rosen
Individual

November 2, 2020

Multicast Source Redundancy in EVPN Networks
draft-skr-bess-evpn-redundant-mcast-source-02

Abstract

EVPN supports intra and inter-subnet IP multicast forwarding. However, EVPN (or conventional IP multicast techniques for that matter) do not have a solution for the case where: a) a given multicast group carries more than one flow (i.e., more than one source), and b) it is desired that each receiver gets only one of the several flows. Existing multicast techniques assume there are no redundant sources sending the same flow to the same IP multicast group, and, in case there were redundant sources, the receiver's application would deal with the received duplicated packets. This document extends the existing EVPN specifications and assumes that IP Multicast source redundancy may exist. It also assumes that, in case two or more sources send the same IP Multicast flows into the tenant domain, the EVPN PEs need to avoid that the receivers get packet duplication by following the described procedures.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on May 6, 2021.

Table of Contents

1. Introduction

Intra and Inter-subnet IP Multicast forwarding are supported in EVPN networks. [I-D.ietf-bess-evpn-igmp-mld-proxy] describes the procedures required to optimize the delivery of IP Multicast flows when Sources and Receivers are connected to the same EVPN BD

(Broadcast Domain), whereas [I-D.ietf-bess-evpn-irb-mcast] specifies the procedures to support Inter-subnet IP Multicast in a tenant network. Inter-subnet IP Multicast means that IP Multicast Source and Receivers of the same multicast flow are connected to different BDs of the same tenant.

[I-D.ietf-bess-evpn-igmp-mld-proxy], [I-D.ietf-bess-evpn-irb-mcast] or conventional IP multicast techniques do not have a solution for the case where a given multicast group carries more than one flow (i.e., more than one source) and it is desired that each receiver gets only one of the several flows. Multicast techniques assume there are no redundant sources sending the same flows to the same IP multicast group, and, in case there were redundant sources, the receiver's application would deal with the received duplicated packets.

As a workaround in conventional IP multicast (PIM or MVPN networks), if all the redundant sources are given the same IP address, each receiver will get only one flow. The reason is that, in conventional IP multicast, (S,G) state is always created by the RP (Rendezvous Point), and sometimes by the Last Hop Router (LHR). The (S,G) state always binds the (S,G) flow to a source-specific tree, rooted at the source IP address. If multiple sources have the same IP address, one may end up with multiple (S,G) trees. However, the way the trees are constructed ensures that any given LHR or RP is on at most one of them. The use of an anycast address assigned to multiple sources may be useful for warm standby redundancy solutions. However, on one hand, it's not really helpful for hot standby redundancy solutions and on the other hand, configuring the same IP address (in particular IPv4 address) in multiple sources may bring issues if the sources need to be reached by IP unicast traffic or if the sources are attached to the same Broadcast Domain.

In addition, in the scenario where several G-sources are attached via EVPN/OISM, there is not necessarily any (S,G) state created for the redundant sources. The LHRs may have only (*,G) state, and there may not be an RP (creating (S,G) state) either. Therefore, this document extends the above two specifications and assumes that IP Multicast source redundancy may exist. It also assumes that, in case two or more sources send the same IP Multicast flows into the tenant domain, the EVPN PEs need to avoid that the receivers get packet duplication.

The solution provides support for Warm Standby (WS) and Hot Standby (HS) redundancy. WS is defined as the redundancy scenario in which the upstream PEs attached to the redundant sources of the same tenant, make sure that only one source of the same flow can send multicast to the interested downstream PEs at the same time. In HS the upstream PEs forward the redundant multicast flows to the

downstream PEs, and the downstream PEs make sure only one flow is forwarded to the interested attached receivers.

1.1. Terminology

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

- PIM: Protocol Independent Multicast.
- MVPN: Multicast Virtual Private Networks.
- OISM: Optimized Inter-Subnet Multicast, as in [I-D.ietf-bess-evpn-irb-mcast].
- Broadcast Domain (BD): an emulated ethernet, such that two systems on the same BD will receive each other's link-local broadcasts. In this document, BD also refers to the instantiation of a Broadcast Domain on an EVPN PE. An EVPN PE can be attached to one or multiple BDs of the same tenant.
- Designated Forwarder (DF): as defined in [RFC7432], an ethernet segment may be multi-homed (attached to more than one PE). An ethernet segment may also contain multiple BDs, of one or more EVIs. For each such EVI, one of the PEs attached to the segment becomes that EVI's DF for that segment. Since a BD may belong to only one EVI, we can speak unambiguously of the BD's DF for a given segment.
- Upstream PE: in this document an Upstream PE is referred to as the EVPN PE that is connected to the IP Multicast source or closest to it. It receives the IP Multicast flows on local ACs (Attachment Circuits).
- Downstream PE: in this document a Downstream PE is referred to as the EVPN PE that is connected to the IP Multicast receivers and gets the IP Multicast flows from remote EVPN PEs.
- G-traffic: any frame with an IP payload whose IP Destination Address (IP DA) is a multicast group G.
- G-source: any system sourcing IP multicast traffic to G.
- SFG: Single Flow Group, i.e., a multicast group address G which represents traffic that contains only a single flow. However,

multiple sources – with the same or different IP – may be transmitting an SFG.

- Redundant G-source: a host or router that transmits an SFG in a tenant network where there are more hosts or routers transmitting the same SFG. Redundant G-sources for the same SFG SHOULD have different IP addresses, although they MAY have the same IP address when in different BDs of the same tenant network. Redundant G-sources are assumed NOT to be "bursty" in this document (typical example are Broadcast TV G-sources or similar).

- P-tunnel: Provider tunnel refers to the type of tree a given upstream EVPN PE uses to forward multicast traffic to downstream PEs. Examples of P-tunnels supported in this document are Ingress Replication (IR), Assisted Replication (AR), Bit Indexed Explicit Replication (BIER), multicast Label Distribution Protocol (mLDP) or Point to Multi-Point Resource Reservation protocol with Traffic Engineering extensions (P2MP RSVP-TE).

- Inclusive Multicast Tree or Inclusive Provider Multicast Service Interface (I-PMSI): defined in [RFC6513], in this document it is applicable only to EVPN and refers to the default multicast tree for a given BD. All the EVPN PEs that are attached to a specific BD belong to the I-PMSI for the BD. The I-PMSI trees are signaled by EVPN Inclusive Multicast Ethernet Tag (IMET) routes.

- Selective Multicast Tree or Selective Provider Multicast Service Interface (S-PMSI): defined in [RFC6513], in this document it is applicable only to EVPN and refers to the multicast tree to which only the interested PEs of a given BD belong to. There are two types of EVPN S-PMSIs:

  * EVPN S-PMSIs that require the advertisement of S-PMSI AD routes from the upstream PE, as in [EVPN-BUM]. The interested downstream PEs join the S-PMSI tree as in [EVPN-BUM].

  * EVPN S-PMSIs that don’t require the advertisement of S-PMSI AD routes. They use the forwarding information of the IMET routes, but upstream PEs send IP Multicast flows only to downstream PEs issuing Selective Multicast Ethernet Tag (SMET) routes for the flow. These S-PMSIs are only supported with the following P-tunnels: Ingress Replication (IR), Assisted Replication (AR) and BIER.

This document also assumes familiarity with the terminology of [RFC7432], [RFC4364], [RFC6513], [RFC6514], [I-D.ietf-bess-evpn-igmp-mld-proxy], [I-D.ietf-bess-evpn-irb-mcast], [EVPN-RT5] and [EVPN-BUM].

1.2. Background on IP Multicast Delivery in EVPN Networks

IP Multicast is all about forwarding a single copy of a packet from a source S to a group of receivers G along a multicast tree. That multicast tree can be created in an EVPN tenant domain where S and the receivers for G are connected to the same BD or different BD. In the former case, we refer to Intra-subnet IP Multicast forwarding, whereas the latter case will be referred to as Inter-subnet IP Multicast forwarding.

1.2.1. Intra-subnet IP Multicast Forwarding

When the source S1 and receivers interested in G1 are attached to the same BD, the EVPN network can deliver the IP Multicast traffic to the receivers in two different ways (Figure 1):

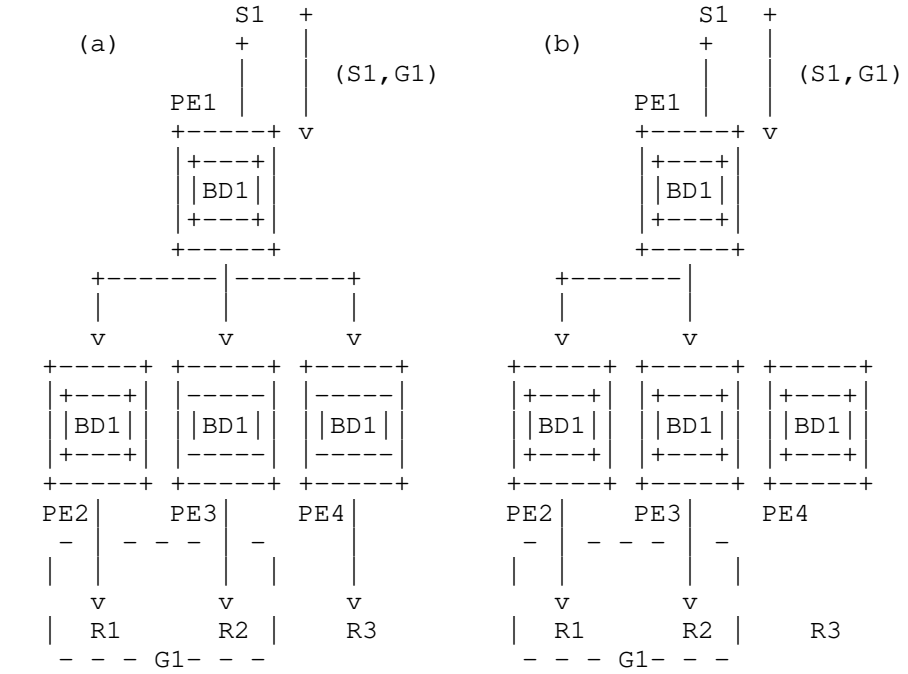


Figure 1: Intra-subnet IP Multicast

Model (a) illustrated in Figure 1 is referred to as "IP Multicast delivery as BUM traffic". This way of delivering IP Multicast traffic does not require any extensions to [RFC7432], however, it sends the IP Multicast flows to non-interested receivers, such as e.g., R3 in Figure 1. In this example, downstream PEs can snoop IGMP/MLD messages from the receivers so that layer-2 multicast state

is created and, for instance, PE4 can avoid sending (S1,G1) to R3, since R3 is not interested in (S1,G1).

Model (b) in Figure 1 uses an S-PMSI to optimize the delivery of the (S1,G1) flow. For instance, assuming PE1 uses IR, PE1 sends (S1,G1) only to the downstream PEs that issued an SMET route for (S1,G1), that is, PE2 and PE3. In case PE1 uses any P-tunnel different than IR, AR or BIER, PE1 will advertise an S-PMSI A-D route for (S1,G1) and PE2/PE2 will join that tree.

Procedures for Model (b) are specified in [I-D.ietf-bess-evpn-igmp-mld-proxy].

1.2.2. Inter-subnet IP Multicast Forwarding

If the source and receivers are attached to different BDs of the same tenant domain, the EVPN network can also use Inclusive or Selective Trees as depicted in Figure 2, models (a) and (b) respectively.

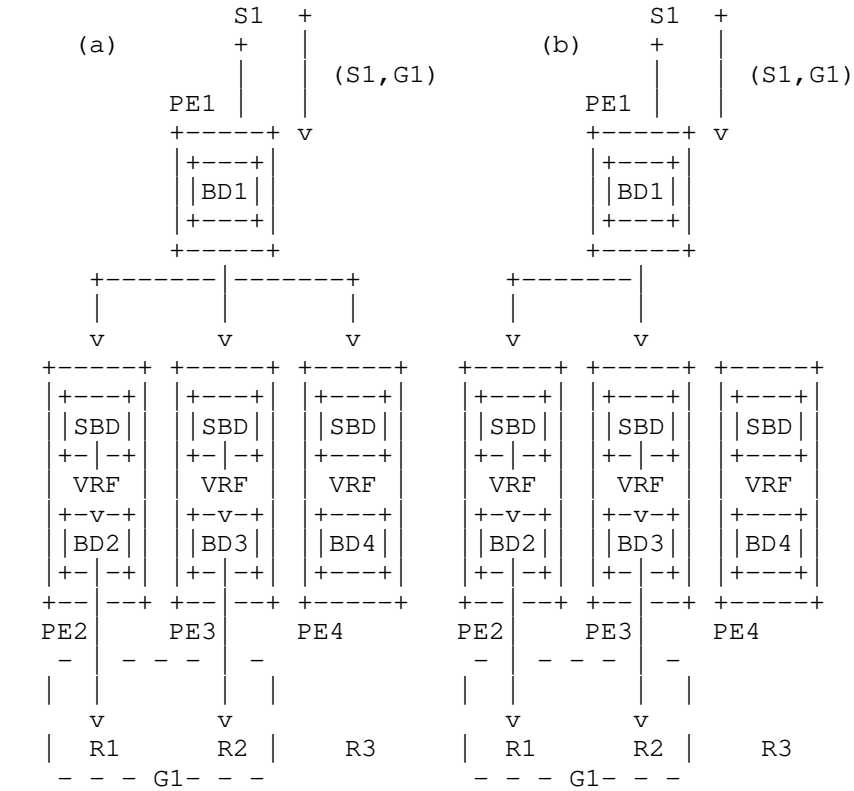


Figure 2: Inter-subnet IP Multicast

[I-D.ietf-bess-evpn-irb-mcast] specifies the procedures to optimize the Inter-subnet Multicast forwarding in an EVPN network. The IP Multicast flows are always sent in the context of the source BD. As described in [I-D.ietf-bess-evpn-irb-mcast], if the downstream PE is not attached to the source BD, the IP Multicast flow is received on the SBD (Supplementary Broadcast Domain), as in the example in Figure 2.

[I-D.ietf-bess-evpn-irb-mcast] supports Inclusive or Selective Multicast Trees, and as explained in Section 1.2.1, the Selective Multicast Trees are setup in a different way, depending on the P-tunnel being used by the source BD. As an example, model (a) in Figure 2 illustrates the use of an Inclusive Multicast Tree for BD1 on PE1. Since the downstream PEs are not attached to BD1, they will all receive (S1,G1) in the context of the SBD and will locally route the flow to the local ACs. Model (b) uses a similar forwarding model, however PE1 sends the (S1,G1) flow in a Selective Multicast Tree. If the P-tunnel is IR, AR or BIER, PE1 does not need to advertise an S-PMSI A-D route.

[I-D.ietf-bess-evpn-irb-mcast] is a superset of the procedures in [I-D.ietf-bess-evpn-igmp-mld-proxy], in which sources and receivers can be in the same or different BD of the same tenant. [I-D.ietf-bess-evpn-irb-mcast] ensures every upstream PE attached to a source will learn of all other PEs (attached to the same Tenant Domain) that have interest in a particular set of flows. This is because the downstream PEs advertise SMET routes for a set of flows with the SBD's Route Target and they are imported by all the Upstream PEs of the tenant. As a result of that, inter-subnet multicasting can be done within the Tenant Domain, without requiring any Rendezvous Points (RP), shared trees, UMH selection or any other complex aspects of conventional multicast routing techniques.

1.3. Multi-Homed IP Multicast Sources in EVPN

Contrary to conventional multicast routing technologies, multi-homing PEs attached to the same source can never create IP Multicast packet duplication if the PEs use a multi-homed Ethernet Segment (ES). Figure 3 illustrates this by showing two multi-homing PEs (PE1 and PE2) that are attached to the same source (S1). We assume that S1 is connected to an all-active ES by a layer-2 switch (SW1) with a Link Aggregation Group (LAG) to PE1 and PE2.

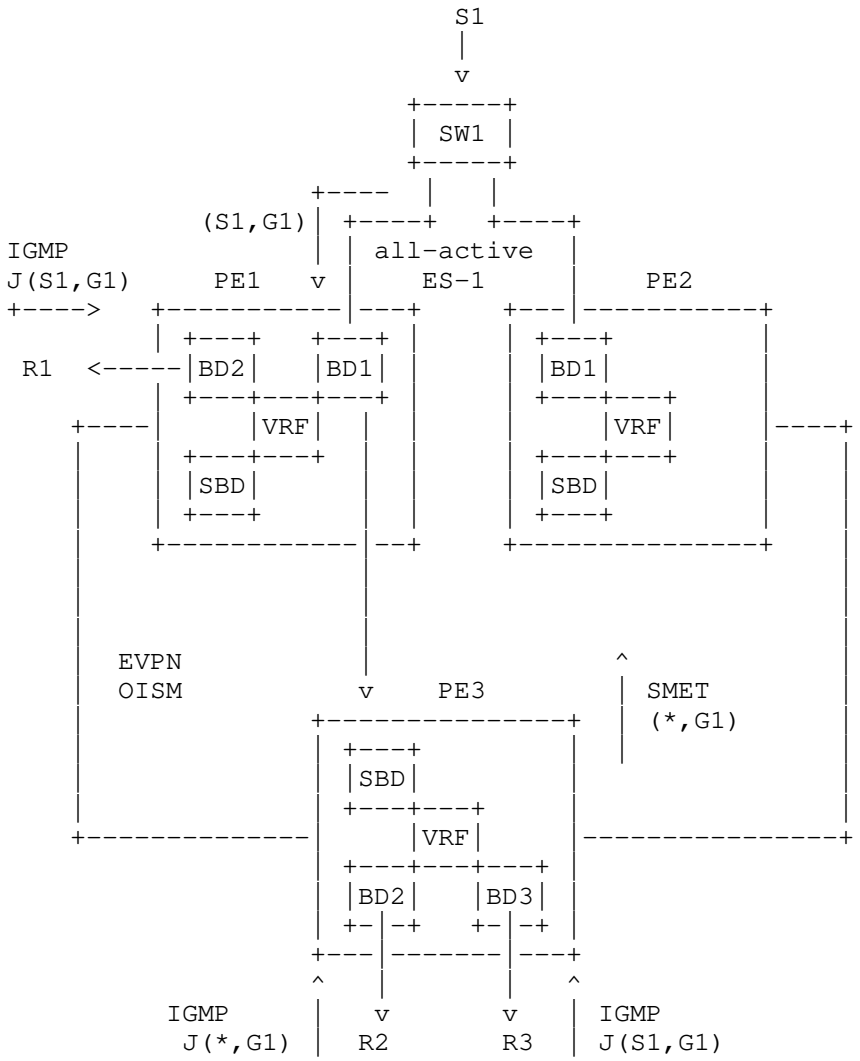


Figure 3: All-active Multi-homing and OISM

When receiving the (S1,G1) flow from S1, SW1 will choose only one link to send the flow, as per [RFC7432]. Assuming PE1 is the receiving PE on BD1, the IP Multicast flow will be forwarded as soon as BD1 creates multicast state for (S1,G1) or (*,G1). In the example of Figure 3, receivers R1, R2 and R3 are interested in the multicast flow to G1. R1 will receive (S1,G1) directly via the IRB interface as per [I-D.ietf-bess-evpn-irb-mcast]. Upon receiving IGMP reports from R2 and R3, PE3 will issue an SMET (*,G1) route that will create state in PE1's BD1. PE1 will therefore forward the IP Multicast flow

to PE3's SBD and PE3 will forward to R2 and R3, as per [I-D.ietf-bess-evpn-irb-mcast] procedures.

When IP Multicast source multi-homing is required, EVPN multi-homed Ethernet Segments MUST be used. EVPN multi-homing guarantees that only one Upstream PE will forward a given multicast flow at the time, avoiding packet duplication at the Downstream PEs. In addition, the SMET route for a given flow creates state in all the multi-homing Upstream PEs. Therefore, in case of failure on the Upstream PE forwarding the flow, the backup Upstream PE can forward the flow immediately.

This document assumes that multi-homing PEs attached to the same source always use multi-homed Ethernet Segments.

1.4. The Need for Redundant IP Multicast Sources in EVPN

While multi-homing PEs to the same IP Multicast G-source provides certain level of resiliency, multicast applications are often critical in the Operator's network and greater level of redundancy is required. This document assumes that:

a. Redundant G-sources for an SFG may exist in the EVPN tenant network. A Redundant G-source is a host or a router that sends an SFG in a tenant network where there is another host or router sending traffic to the same SFG.

b. Those redundant G-sources may be in the same BD or different BDs of the tenant. There must not be restrictions imposed on the location of the receiver systems either.

c. The redundant G-sources can be single-homed to only one EVPN PE or multi-homed to multiple EVPN PEs.

d. The EVPN PEs must avoid duplication of the same SFG on the receiver systems.

2. Solution Overview

An SFG is represented as (*,G) if any source that issues multicast traffic to G is a redundant G-source. Alternatively, this document allows an SFG to be represented as (S,G), where S is a prefix of any length. In this case, a source is considered a redundant G-source for the SFG if it is contained in the prefix. This document allows variable length prefixes in the Sources advertised in S-PMSI A-D routes only for the particular application of redundant G-sources.

There are two redundant G-source solutions described in this document:

- Warm Standby (WS) Solution
- Hot Standby (HS) Solution

The WS solution is considered an upstream-PE-based solution (since downstream PEs do not participate in the procedures), in which all the upstream PEs attached to redundant G-sources for an SFG represented by (*,G) or (S,G) will elect a "Single Forwarder" (SF) among themselves. Once a SF is elected, the upstream PEs add an Reverse Path Forwarding (RPF) check to the (*,G) or (S,G) state for the SFG:

- A non-SF upstream PE discards any (*,G)/(S,G) packets received over a local AC.
- The SF accepts and forwards any (*,G)/(S,G) packets it receives over a single local AC (for the SFG). In case (*,G)/(S,G) packets for the SFG are received over multiple local ACs, they will be discarded in all the local ACs but one. The procedure to choose the local AC that accepts packets is a local implementation matter.

A failure on the SF will result in the election of a new SF. The Election requires BGP extensions on the existing EVPN routes. These extensions and associated procedures are described in Section 3 and Section 4 respectively.

In the HS solution the downstream PEs are the ones avoiding the SFG duplication. The upstream PEs are aware of the locally attached G-sources and add a unique Ethernet Segment Identifier label (ESI-label) per SFG to the SFG packets forwarded to downstream PEs. The downstream PEs pull the SFG from all the upstream PEs attached to the redundant G-sources and avoid duplication on the receiver systems by adding an RPF check to the (*,G) state for the SFG:

- A downstream PE discards any (*,G) packets it receives from the "wrong G-source".
- The wrong G-source is identified in the data path by an ESI-label that is different than the ESI-label used for the selected G-source.
- Note that the ESI-label is used here for "ingress filtering" (at the egress/downstream PE) as opposed to the [RFC7432] "egress filtering" (at the egress/downstream PE) used in the split-horizon

procedures. In [RFC7432] the ESI-label indicates what egress ACs must be skipped when forwarding BUM traffic to the egress. In this document, the ESI-label indicates what ingress traffic must be discarded at the downstream PE.

The use of ESI-labels for SFGs forwarded by upstream PEs require some control plane and data plane extensions in the procedures used by [RFC7432] for multi-homing. Upon failure of the selected G-source, the downstream PE will switch over to a different selected G-source, and will therefore change the RPF check for the (*,G) state. The extensions and associated procedures are described in Section 3 and Section 5 respectively.

An operator should use the HS solution if they require a fast fail-over time and the additional bandwidth consumption is acceptable (SFG packets are received multiple times on the downstream PEs). Otherwise the operator should use the WS solution, at the expense of a slower fail-over time in case of a G-source or upstream PE failure. Besides bandwidth efficiency, another advantage of the WS solution is that only the upstream PEs attached to the redundant G-sources for the same SFG need to be upgraded to support the new procedures.

This document does not impose the support of both solutions on a system. If one solution is supported, the support of the other solution is OPTIONAL.

3. BGP EVPN Extensions

This document makes use of the following BGP EVPN extensions:

1. SFG flag in the Multicast Flags Extended Community

   The Single Flow Group (SFG) flag is a new bit requested to IANA out of the registry Multicast Flags Extended Community Flag Values. This new flag is set for S-PMSI A-D routes that carry a (*,G)/(S,G) SFG in the NLRI.

2. ESI Label Extended Community is used in S-PMSI A-D routes

   The HS solution requires the advertisement of one or more ESI Label Extended Communities [RFC7432] that encode the Ethernet Segment Identifier(s) associated to an S-PMSI A-D (*,G)/(S,G) route that advertises the presence of an SFG. Only the ESI Label value in the extended community is relevant to the procedures in this document. The Flags field in the extended community will be advertised as 0x00 and ignored on reception. [RFC7432] specifies that the ESI Label Extended Community is advertised along with the A-D per ES route. This documents extends the use of this

extended community so that it can be advertised multiple times (with different ESI values) along with the S-PMSI A-D route.

4. Warm Standby (WS) Solution for Redundant G-Sources

The general procedure is described as follows:

1. Configuration of the upstream PEs

   Upstream PEs (possibly attached to redundant G-sources) need to be configured to know which groups are carrying only flows from redundant G-sources, that is, the SFGs in the tenant domain. They will also be configured to know which local BDs may be attached to a redundant G-source. The SFGs can be configured for any source, E.g., SFG for "*", or for a prefix that contains multiple sources that will issue the same SFG, i.e., "10.0.0.0/30". In the latter case sources 10.0.0.1 and 10.0.0.2 are considered as Redundant G-sources, whereas 10.0.0.10 is not considered a redundant G-source for the same SFG.

   As an example:

   * PE1 is configured to know that G1 is an SFG for any source and redundant G-sources for G1 may be attached to BD1 or BD2.

   * Or PE1 can also be configured to know that G1 is an SFG for the sources contained in 10.0.0.0/30, and those redundant G-sources may be attached to BD1 or BD2.

2. Signaling the location of a G-source for a given SFG

   Upon receiving G-traffic for a configured SFG on a BD, an upstream PE configured to follow this procedure, e.g., PE1:

   * Originates an S-PMSI A-D (*,G)/(S,G) route for the SFG. An (*,G) route is advertised if the SFG is configured for any source, and an (S,G) route is advertised (where the Source can have any length) if the SFG is configured for a prefix.

   * The S-PMSI A-D route is imported by all the PEs attached to the tenant domain. In order to do that, the route will use the SBD-RT (Supplementary Broadcast Domain Route-Target) in addition to the BD-RT of the BD over which the G-traffic is received. The route SHOULD also carry a DF Election Extended Community (EC) and a flag indicating that it conveys an SFG. The DF Election EC and its use is specified in [RFC8584].

* The above S-PMSI A-D route MAY be advertised with or without PMSI Tunnel Attribute (PTA):

  + With no PTA if an I-PMSI or S-PMSI A-D with IR/AR/BIER are to be used.

  + With PTA in any other case.

* The S-PMSI A-D route is triggered by the first packet of the SFG and withdrawn when the flow is not received anymore. Detecting when the G-source is no longer active is a local implementation matter. The use of a timer is RECOMMENDED. The timer is started when the traffic to G1 is not received. Upon expiration of the timer, the PE will withdraw the route

3. Single Forwarder (SF) Election

   If the PE with a local G-source receives one or more S-PMSI A-D routes for the same SFG from a remote PE, it will run a Single Forwarder (SF) Election based on the information encoded in the DF Election EC. Two S-PMSI A-D routes are considered for the same SFG if they are advertised for the same tenant, and their Multicast Source Length, Multicast Source, Multicast Group Length and Multicast Group fields match.

   1. A given DF Alg can only be used if all the PEs running the DF Alg have consistent input. For example, in an OISM network, if the redundant G-sources for an SFG are attached to BDs with different Ethernet Tags, the Default DF Election Alg MUST NOT be used.

   2. In case the there is a mismatch in the DF Election Alg or capabilities advertised by two PEs competing for the SF, the lowest PE IP address (given by the Originator Address in the S- PMSI A-D route) will be used as a tie-breaker.

4. RPF check on the PEs attached to a redundant G-source

   All the PEs with a local G-source for the SFG will add an RPF check to the (*,G)/(S,G) state for the SFG. That RPF check depends on the SF Election result:

   1. The non-SF PEs discard any (*,G)/(S,G) packets for the SFG received over a local AC.

   2. The SF accepts any (*,G)/(S,G) packets for the SFG it receives over one (and only one) local AC.

The solution above provides redundancy for SFGs and it does not require an upgrade of the downstream PEs (PEs where there is certainty that no redundant G-sources are connected). Other G-sources for non-SFGs may exist in the same tenant domain. This document does not change the existing procedures for non-SFG G-sources.

The redundant G-sources can be single-homed or multi-homed to a BD in the tenant domain. Multi-homing does not change the above procedures.

Section 4.1 and Section 4.2 show two examples of the WS solution.

4.1. WS Example in an OISM Network

Figure 4 illustrates an example in which S1 and S2 are redundant G-sources for the SFG (*,G1).

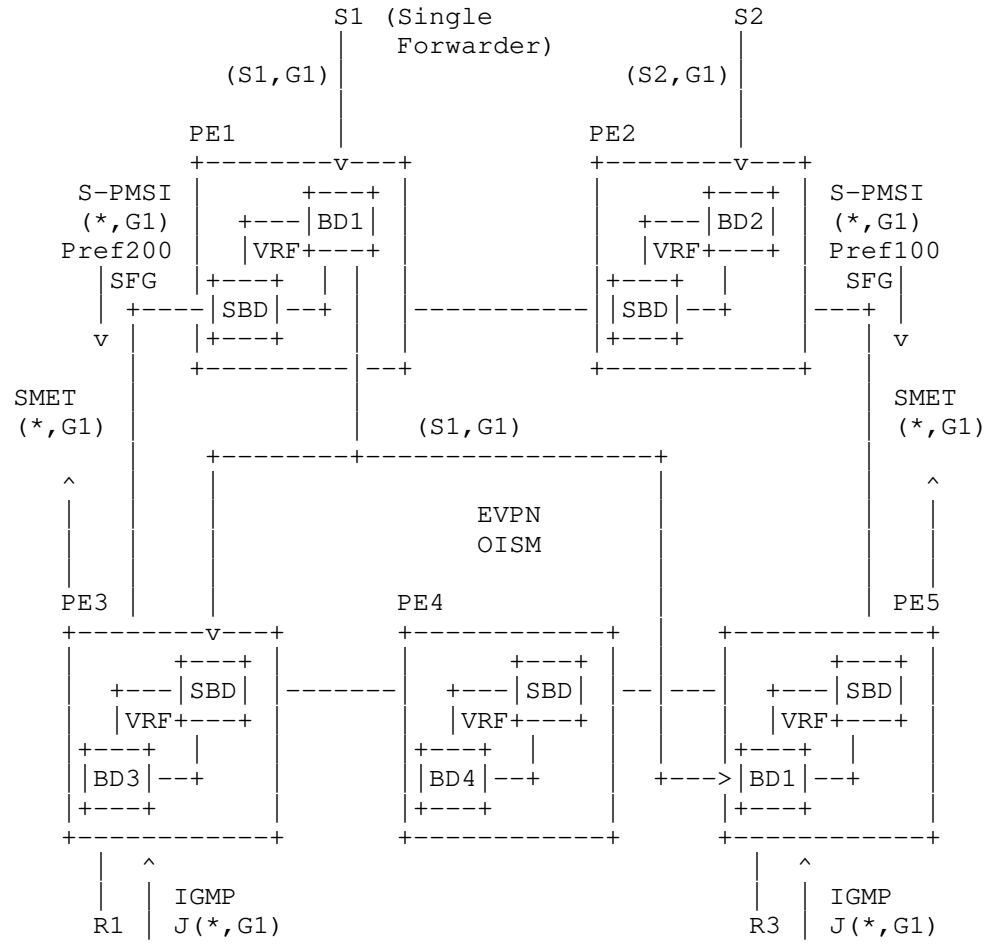


Figure 4: WS Solution for Redundant G-Sources

The WS solution works as follows:

1. Configuration of the upstream PEs, PE1 and PE2

   PE1 and PE2 are configured to know that G1 is an SFG for any source and redundant G-sources for G1 may be attached to BD1 or BD2, respectively.

2. Signaling the location of S1 and S2 for (*,G1)

   Upon receiving (S1,G1) traffic on a local AC, PE1 and PE2 originate S-PMSI A-D (*,G1) routes with the SBD-RT, DF Election

Extended Community (EC) and a flag indicating that it conveys an SFG.

3. Single Forwarder (SF) Election

   Based on the DF Election EC content, PE1 and PE2 elect an SF for (*,G1). Assuming both PEs agree on e.g., Preference based Election as the algorithm to use [DF-PREF], and PE1 has a higher preference, PE1 becomes the SF for (*,G1).

4. RPF check on the PEs attached to a redundant G-source

   A. The non-SF, PE2, discards any (*,G1) packets received over a local AC.

   B. The SF, PE1 accepts (*,G1) packets it receives over one (and only one) local AC.

The end result is that, upon receiving reports for (*,G1) or (S,G1), the downstream PEs (PE3 and PE5) will issue SMET routes and will pull the multicast SFG from PE1, and PE1 only. Upon a failure on S1, the AC connected to S1 or PE1 itself will trigger the S-PMSI A-D (*,G1) withdrawal from PE1 and PE2 will be promoted to SF.

4.2. WS Example in a Single-BD Tenant Network

Figure 5 illustrates an example in which S1 and S2 are redundant G-sources for the SFG (*,G1), however, now all the G-sources and receivers are connected to the same BD1 and there is no SBD.

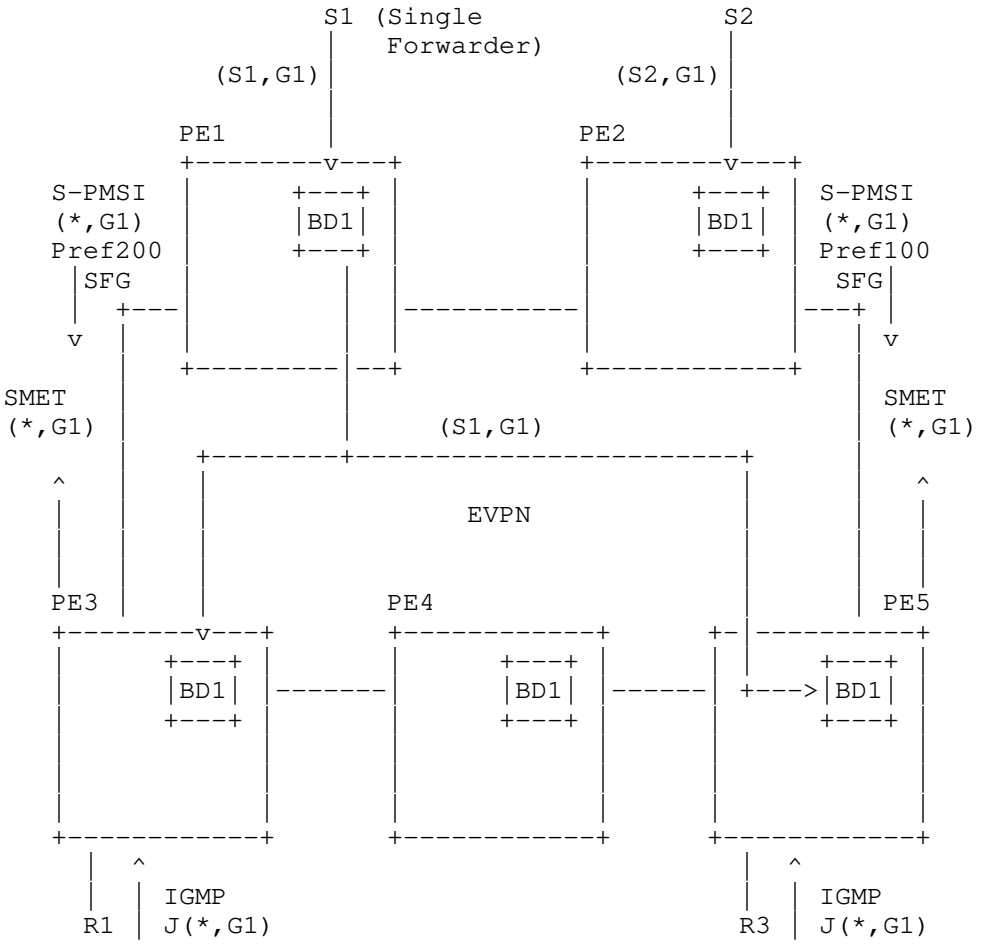


Figure 5: WS Solution for Redundant G-Sources in the same BD

The same procedure as in Section 4.1 is valid here, being this a sub-case of the one in Section 4.1. Upon receiving traffic for the SFG G1, PE1 and PE2 advertise the S-PMSI A-D routes with BD1-RT only, since there is no SBD.

5. Hot Standby (HS) Solution for Redundant G-Sources

If fast-failover is required upon the failure of a G-source or PE attached to the G-source and the extra bandwidth consumption in the tenant network is not an issue, the HS solution should be used. The procedure is as follows:

1. Configuration of the PEs

As in the WS case, the upstream PEs where redundant G-sources may exist need to be configured to know which groups (for any source or a prefix containing the intended sources) are carrying only flows from redundant G-sources, that is, the SFGs in the tenant domain.

In addition (and this is not done in WS mode), the individual redundant G-sources for an SFG need to be associated with an Ethernet Segment (ES) on the upstream PEs. This is irrespective of the redundant G-source being multi-homed or single-homed. Even for single-homed redundant G-sources the HS procedure relies on the ESI labels for the RPF check on downstream PEs. The term "S-ESI" is used in this document to refer to an ESI associated to a redundant G-source.

Contrary to what is specified in the WS method (that is transparent to the downstream PEs), the support of the HS procedure is required not only on the upstream PEs but also on all downstream PEs connected to the receivers in the tenant network. The downstream PEs do not need to be configured to know the connected SFGs or their ESIs, since they get that information from the upstream PEs. The downstream PEs will locally select an ESI for a given SFG, and will program an RPF check to the (*,G)/(S,G) state for the SFG that will discard (*,G)/(S,G) packets from the rest of the ESIs. The selection of the ESI for the SFG is based on local policy.

2. Signaling the location of a G-source for a given SFG and its association to the local ESIs

   Based on the configuration in step 1, an upstream PE configured to follow the HS procedures:

   A. Advertises an S-PMSI A-D (*,G)/(S,G) route per each configured SFG. These routes need to be imported by all the PEs of the tenant domain, therefore they will carry the BD-RT and SBD-RT (if the SBD exists). The route also carries the ESI Label Extended Communities needed to convey all the S-ESIs associated to the SFG in the PE.

   B. The S-PMSI A-D route will convey a PTA in the same cases as in the WS procedure.

   C. The S-PMSI A-D (*,G)/(S,G) route is triggered by the configuration of the SFG and not by the reception of G-traffic.

3. Distribution of DCB (Domain-wide Common Block) ESI-labels and G-source ES routes

   An upstream PE advertises the corresponding ES, A-D per EVI and A-D per ES routes for the local S-ESIs.

   A. ES routes are used for regular DF Election for the S-ES. This document does not introduce any change in the procedures related to the ES routes.

   B. The A-D per EVI and A-D per ES routes MUST include the SBD-RT since they have to be imported by all the PEs in the tenant domain.

   C. The A-D per ES routes convey the S-ESI labels that the downstream PEs use to add the RPF check for the (*,G)/(S,G) associated to the SFGs. This RPF check requires that all the packets for a given G-source are received with the same S-ESI label value on the downstream PEs. For example, if two redundant G-sources are multi-homed to PE1 and PE2 via S-ES-1 and S-ES-2, PE1 and PE2 MUST allocate the same ESI label "Lx" for S-ES-1 and they MUST allocate the same ESI label "Ly" for S-ES-2. In addition, Lx and Ly MUST be different. These ESI labels are Domain-wide Common Block (DCB) labels and follow the allocation procedures in [I-D.zzhang-bess-mvpn-evpn-aggregation-label].

4. Processing of A-D per ES/EVI routes and RPF check on the downstream PEs

   The A-D per ES/EVI routes are received and imported in all the PEs in the tenant domain. The processing of the A-D per ES/EVI routes on a given PE depends on its configuration:

   A. The PEs attached to the same BD of the BD-RT that is included in the A-D per ES/EVI routes will process the routes as in [RFC7432] and [RFC8584]. If the receiving PE is attached to the same ES as indicated in the route, [RFC7432] split-horizon procedures will be followed and the DF Election candidate list may be modified as in [RFC8584] if the ES supports the AC-DF capability.

   B. The PEs that are not attached to the BD-RT but are attached to the SBD of the received SBD-RT, will import the A-D per ES/EVI routes and use them for redundant G-source mass withdrawal, as explained later.

C. Upon importing A-D per ES routes corresponding to different S-ESes, a PE MUST select a primary S-ES and add an RPF check to the (*,G)/(S,G) state in the BD or SBD. This RPF check will discard all ingress packets to (*,G)/(S,G) that are not received with the ESI-label of the primary S-ES. The selection of the primary S-ES is a matter of local policy.

5. G-traffic forwarding for redundant G-sources and fault detection

Assuming there is (*,G) or (S,G) state for the SFG with OIF (Ouput Interface) list entries associated to remote EVPN PEs, upon receiving G-traffic on a S-ES, the upstream PE will add a S-ESI label at the bottom of the stack before forwarding the traffic to the remote EVPN PEs. This label is allocated from a DCB as described in step 3. If P2MP or BIER PMSIs are used, this is not adding any new data path procedures on the upstream PEs (except that the ESI-label is allocated from a DCB). However, if IR/AR are used, this document extends the [RFC7432] procedures by pushing the S-ESI labels not only on packets sent to the PEs that shared the ES but also to the rest of the PEs in the tenant domain. This allows the downstream PEs to receive all the multicast packets from the redundant G-sources with a S-ESI label (irrespective of the PMSI type and the local ESes), and discard any packet that conveys a S-ESI label different from the primary S-ESI label (that is, the label associated to the selected primary S-ES), as discussed in step 4.

If the last A-D per EVI or the last A-D per ES route for the primary S-ES is withdrawn, the downstream PE will immediately select a new primary S-ES and will change the RPF check. Note that if the S-ES is re-used for multiple tenant domains by the upstream PEs, the withdrawal of all the A-D per-ES routes for a S-ES provides a mass withdrawal capability that makes a downstream PE to change the RPF check in all the tenant domains using the same S-ES.

The withdrawal of the last S-PMSI A-D route for a given (*,G)/(S,G) that represents a SFG SHOULD make the downstream PE remove the S-ESI label based RPF check on (*,G)/(S,G).

5.1. Use of BFD in the HS Solution

In addition to using the state of the A-D per EVI, A-D per ES or S-PMSI A-D routes to modify the RPF check on (*,G)/(S,G) as discussed in Section 5, Bidirectional Forwarding Detection (BFD) protocol MAY be used to find the status of the multipoint tunnels used to forward the SFG from the redundant G-sources.

The BGP-BFD Attribute is advertised along with the S-PMSI A-D or IMET routes (depending on whether I-PMSI or S-PMSI trees are used) and the procedures described in [EVPN-BFD] are used to bootstrap multipoint BFD sessions on the downstream PEs.

5.2. HS Example in an OISM Network

Figure 6 illustrates the HS model in an OISM network. Consider S1 and S2 are redundant G-sources for the SFG (*,G1) in BD1 (any source using G1 is assumed to transmit an SFG). S1 and S2 are (all-active) multi-homed to upstream PEs, PE1 and PE2. The receivers are attached to downstream PEs, PE3 and PE5, in BD3 and BD1, respectively. S1 and S2 are assumed to be connected by a LAG to the multi-homing PEs, and the multicast traffic can use the link to either upstream PE. The diagram illustrates how S1 sends the G-traffic to PE1 and PE1 forwards to the remote interested downstream PEs, whereas S2 sends to PE2 and PE2 forwards further. In this HS model, the interested downstream PEs will get duplicate G-traffic from the two G-sources for the same SFG. While the diagram shows that the two flows are forwarded by different upstream PEs, the all-active multi-homing procedures may cause that the two flows come from the same upstream PE. Therefore, finding out the upstream PE for the flow is not enough for the downstream PEs to program the required RPF check to avoid duplicate packets on the receiver.

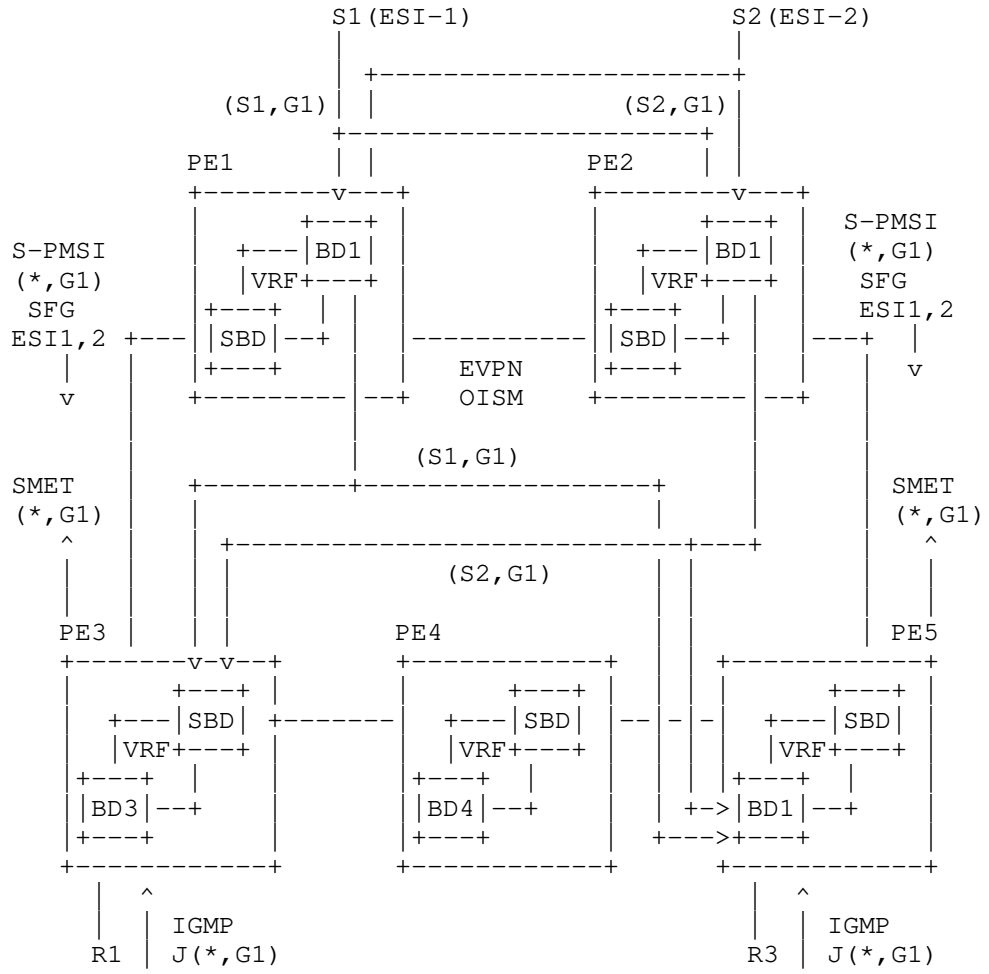


Figure 6: HS Solution for Multi-homed Redundant G-Sources in OISM

In this scenario, the HS solution works as follows:

1. Configuration of the upstream PEs, PE1 and PE2

   PE1 and PE2 are configured to know that G1 is an SFG for any source (a source prefix length could have been configured instead) and the redundant G-sources for G1 use S-ESIs ESI-1 and ESI-2 respectively. Both ESes are configured in both PEs and the ESI value can be configured or auto-derived. The ESI-label values are allocated from a DCB [I-D.zzhang-bess-mvpn-evpn-aggregation-label] and are configured

either locally or by a centralized controller. We assume ESI-1 is configured to use ESI-label-1 and ESI-2 to use ESI-label-2.

The downstream PEs, PE3, PE4 and PE5 are configured to support HS mode and select the G-source with e.g., lowest ESI value.

2. PE1 and PE2 advertise S-PMSI A-D (*,G1) and ES/A-D per ES/EVI routes

   Based on the configuration of step 1, PE1 and PE2 advertise an S-PMSI A-D (*,G1) route each. The route from each of the two PEs will include TWO ESI Label Extended Communities with ESI-1 and ESI-2 respectively, as well as BD1-RT plus SBD-RT and a flag that indicates that (*,G1) is an SFG.

   In addition, PE1 and PE2 advertise ES and A-D per ES/EVI routes for ESI-1 and ESI-2. The A-D per ES and per EVI routes will include the SBD-RT so that they can be imported by the downstream PEs that are not attached to BD1, e.g., PE3 and PE4. The A-D per ES routes will convey ESI-label-1 for ESI-1 (on both PEs) and ESI-label-2 for ESI-2 (also on both PEs).

3. Processing of A-D per ES/EVI routes and RPF check

   PE1 and PE2 received each other’s ES and A-D per ES/EVI routes. Regular [RFC7432] [RFC8584] procedures will be followed for DF Election and programming of the ESI-labels for egress split-horizon filtering. PE3/PE4 import the A-D per ES/EVI routes in the SBD. Since PE3 has created a (*,G1) state based on local interest, PE3 will add an RPF check to (*,G1) so that packets coming with ESI-label-2 are discarded (lowest ESI value is assumed to give the primary S-ES).

4. G-traffic forwarding and fault detection

   PE1 receives G-traffic (S1,G1) on ES-1 that is forwarded within the context of BD1. Irrespective of the tunnel type, PE1 pushes ESI-label-1 at the bottom of the stack and the traffic gets to PE3 and PE5 with the mentioned ESI-label (PE4 has no local interested receivers). The G-traffic with ESI-label-1 passes the RPF check and it is forwarded to R1. In the same way, PE2 sends (S2,G1) with ESI-label-2, but this G-traffic does not pass the RPF check and gets discarded at PE3/PE5.

   If the link from S1 to PE1 fails, S1 will forward the (S1,G1) traffic to PE2 instead. PE1 withdraws the ES and A-D routes for ESI-1. Now both flows will be originated by PE2, however the RPF checks don’t change in PE3/PE5.

If subsequently, the link from S1 to PE2 fails, PE2 also withdraws the ES and A-D routes for ESI-1. Since PE3 and PE5 have no longer A-D per ES/EVI routes for ESI-1, they immediately change the RPF check so that packets with ESI-label-2 are now accepted.

Figure 7 illustrates a scenario where S1 and S2 are single-homed to PE1 and PE2 respectively. This scenario is a sub-case of the one in Figure 6. Now ES-1 only exists in PE1, hence only PE1 advertises the A-D per ES/EVI routes for ESI-1. Similarly, ES-2 only exists in PE2 and PE2 is the only PE advertising A-D routes for ESI-2. The same procedures as in Figure 6 applies to this use-case.

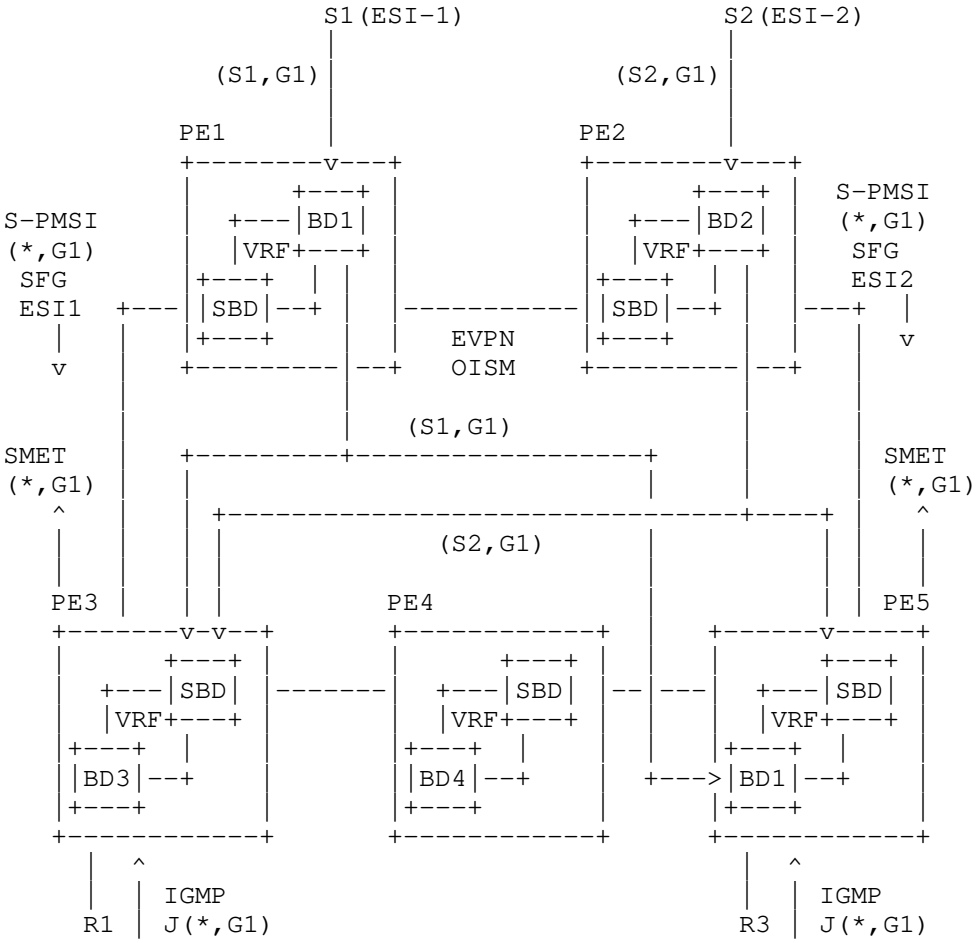


Figure 7: HS Solution for single-homed Redundant G-Sources in OISM

5.3. HS Example in a Single-BD Tenant Network

Irrespective of the redundant G-sources being multi-homed or single-homed, if the tenant network has only one BD, e.g., BD1, the procedures of Section 5.2 still apply, only that routes do not include any SBD-RT and all the procedures apply to BD1 only.

6. Security Considerations

The same Security Considerations described in [I-D.ietf-bess-evpn-irb-mcast] are valid for this document.

From a security perspective, out of the two methods described in this document, the WS method is considered lighter in terms of control plane and therefore its impact is low on the processing capabilities of the PEs. The HS method adds more burden on the control plane of all the PEs of the tenant with sources and receivers.

7. IANA Considerations

IANA is requested to allocate a Bit in the Multicast Flags Extended Community to indicate that a given (*,G) or (S,G) in an S-PMSI A-D route is associated with an SFG.

8. References

8.1. Normative References

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <https://www.rfc-editor.org/info/rfc6513>.

[RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, DOI 10.17487/RFC6514, February 2012, <https://www.rfc-editor.org/info/rfc6514>.

[I-D.ietf-bess-evpn-igmp-mld-proxy] Sajassi, A., Thoria, S., Patel, K., Drake, J., and W. Lin, "IGMP and MLD Proxy for EVPN", draft-ietf-bess-evpn-igmp-mld-proxy-05 (work in progress), April 2020.

[I-D.ietf-bess-evpn-irb-mcast]
Lin, W., Zhang, Z., Drake, J., Rosen, E., Rabadan, J., and A. Sajassi, "EVPN Optimized Inter-Subnet Multicast (OISM) Forwarding", draft-ietf-bess-evpn-irb-mcast-05 (work in progress), October 2020.

[RFC8584] Rabadan, J., Ed., Mohanty, S., Ed., Sajassi, A., Drake, J., Nagaraj, K., and S. Sathappan, "Framework for Ethernet VPN Designated Forwarder Election Extensibility", RFC 8584, DOI 10.17487/RFC8584, April 2019, <https://www.rfc-editor.org/info/rfc8584>.

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

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

[I-D.zzhang-bess-mvpn-evpn-aggregation-label]
Zhang, Z., Rosen, E., Lin, W., Li, Z., and I. Wijnands, "MVPN/EVPN Tunnel Aggregation with Common Labels", draft-zzhang-bess-mvpn-evpn-aggregation-label-01 (work in progress), April 2018.

8.2. Informative References

[EVPN-RT5]
Rabadan, J., Henderickx, W., Drake, J., Lin, W., and A. Sajassi, "IP Prefix Advertisement in EVPN", internet-draft ietf-bess-evpn-prefix-advertisement-11.txt, May 2018.

[EVPN-BUM]
Zhang, Z., Lin, W., Rabadan, J., and K. Patel, "Updates on EVPN BUM Procedures", internet-draft ietf-bess-evpn-bum-procedure-updates-06, June 2019.

[DF-PREF] Rabadan, J., Sathappan, S., Przygienda, T., Lin, W., Drake, J., Sajassi, A., and S. Mohanty, "Preference-based EVPN DF Election", internet-draft ietf-bess-evpn-pref-df-04.txt, June 2019.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[EVPN-BFD]
Govindan, V., Mallik, M., Sajassi, A., and G. Mirsky, "Fault Management for EVPN networks", internet-draft ietf-bess-evpn-bfd-01.txt, October 2020.

Appendix A. Acknowledgments

The authors would like to thank Mankamana Mishra and Ali Sajassi for their review and valuable comments.

Appendix B. Contributors

Authors' Addresses

Jorge Rabadan (editor)
Nokia
777 Middlefield Road
Mountain View, CA 94043
USA

Email: jorge.rabadan@nokia.com

Jayant Kotalwar
Nokia
701 E. Middlefield Road
Mountain View, CA 94043 USA

Email: jayant.kotalwar@nokia.com

Senthil Sathappan
Nokia
701 E. Middlefield Road
Mountain View, CA 94043 USA

Email: senthil.sathappan@nokia.com

Zhaohui Zhang
Juniper Networks

Email: zzhang@juniper.net

Wen Lin
Juniper Networks

Email: wlin@juniper.net

Eric C. Rosen
Individual

Email: erosen52@gmail.com

BESS Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 24, 2020

Q. Wu, Ed.
Huawei
M. Boucadair, Ed.
Orange
O. Gonzalez de Dios
Telefonica
B. Wen
Comcast
C. Liu
China Unicom
H. Xu
China Telecom

April 22, 2020

A YANG Model for Network and VPN Service Performance Monitoring
draft-www-bess-yang-vpn-service-pm-06

Abstract

The data model defined in RFC8345 introduces vertical layering relationships between networks that can be augmented to cover network/service topologies. This document defines a YANG model for both Network Performance Monitoring and VPN Service Performance Monitoring that can be used to monitor and manage network performance on the topology at higher layer or the service topology between VPN sites.

This model is designed as an augmentation to the network topology YANG data model defined in RFC8345.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on October 24, 2020.

Table of Contents

1. Introduction

[RFC8345] defines a YANG data model for network/service topologies and inventories. The service topology described in [RFC8345] includes the virtual topology for a service layer above Layer 1 (L1), Layer 2 (L2), and Layer 3 (L3). This service topology has the generic topology elements of node, link, and terminating point. One typical example of a service topology is described in Figure 3 of

[RFC8345]: two VPN service topologies instantiated over a common L3 topology. Each VPN service topology is mapped onto a subset of nodes from the common L3 topology.

Three types of VPN service topologies are supported in [RFC8299]: "any to any", "hub and spoke", and "hub and spoke disjoint". These VPN topology types can be used to describe how VPN sites communicate with each other.

This document defines a YANG Model for both Network Performance Monitoring and VPN Service Performance Monitoring (see Section 2.2.4 of [RFC4176]) that can be used to monitor and manage network Performance on the topology at higher layer or the service topology between VPN sites.

The model is designed as an augmentation to the network topology YANG data model defined in [RFC8345].

2. Terminology

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

Tree diagrams used in this document follow the notation defined in [RFC8340].

3. Network and VPN Service Assurance Module

The module defined in this document is a Network and VPN Service assurance module that can be used to monitor and manage the network performance on the topology at higher layer or the service topology between VPN sites and it is an augmentation to the "ietf-network" and "ietf-network-topology" YANG data model [RFC8345].

The performance monitoring data is augmented to service topology as shown in Figure 1.

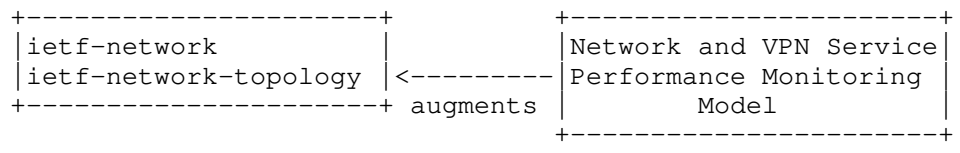


Figure 1: Module Augmentation

4. Layering Relationship Between Multiple Layers of Topology

The data model defined in [RFC8345] can describe vertical layering relationships between networks. That model can be augmented to cover network/service topologies.

Figure 2 illustrates an example of a topology mapping between the VPN service topology and an underlying network:

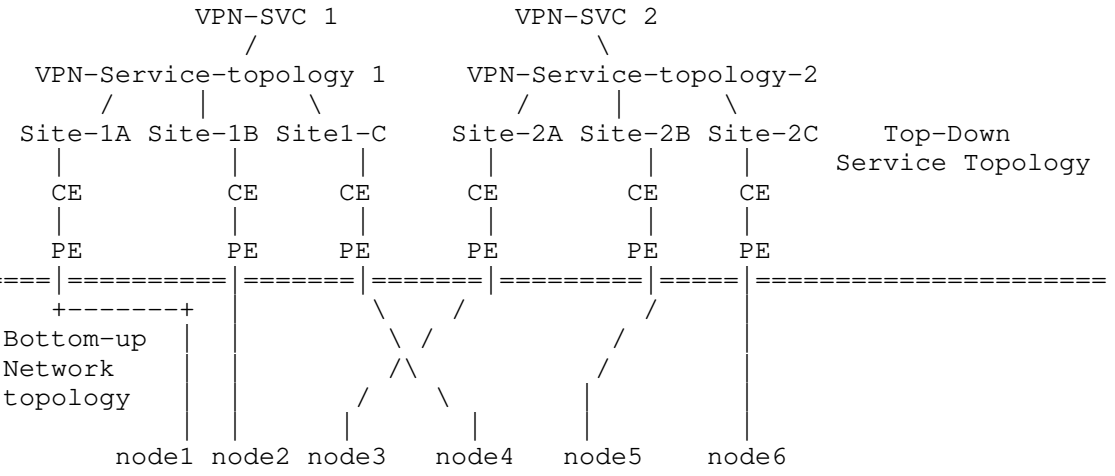


Figure 2: Example of topology mapping between VPN Service Topo and Underlying network

As shown in Figure 2, two VPN services topologies are both built on top of one common underlying physical network:

- VPN-SVC 1: supporting "hub-spoke" communications for Customer 1 connecting the customer's access at 3 sites. Site-1A, Site-1B, and Site-1C are connected to PEs that are mapped to nodes 1, 2, and 3 in the underlying physical network.

  Site-1 A plays the role of hub while Site-2 B and C plays the role of spoke.

- VPN-SVC 2: supporting "hub-spoke disjoint" communications for Customer 2 connecting the customer's access at 3 sites. Site-2A, Site-2B, and Site-2C are connected to PEs that are mapped to nodes 4, 5, and 6 in the underlying physical network.

  Site-2 A and B play the role of hub while Site-2 C plays the role of spoke.

5. Some Model Usage Guidelines

An SP must be able to manage the capabilities and characteristics of the network/VPN services when Network connection is established or VPN sites are setup to communicate with each other.

5.1. Performance Monitoring Data Source

As described in Section 4, once the mapping between the VPN Service topology and the underlying physical network has been setup, the performance monitoring data per link in the underlying network can be collected using network performance measurement method such as MPLS Loss and Delay Measurement [RFC6374].

The performance monitoring information reflecting the quality of the Network or VPN service such as end to end network performance data between source node and destination node in the network or between VPN sites can be aggregated or calculated using, for example, PCEP solution [RFC8233] [RFC7471] [RFC7810] [RFC8571] or LMAP [RFC8194].

The information can be fed into data source such as the management system or network devices. The measurement interval and report interval associated with these performance data usually depends on configuration parameters.

5.2. Retrieval via Pub/Sub Mechanism

Some applications such as service-assurance applications, which must maintain a continuous view of operational data and state, can use subscription model [I-D.ietf-netconf-yang-push] to subscribe to the specific Network performance data or VPN service performance data they are interested in, at the data source.

The data source can then use the Network and VPN service assurance model defined in this document and the YANG Push model [I-D.ietf-netconf-yang-push] to distribute specific telemetry data to target recipients.

5.3. On demand Retrieval via RPC Model

To obtain a snapshot of a large amount of performance data from a network element (including network controllers), service-assurance applications may use polling-based methods such as RPC model to fetch performance data on demand.

6. Data Model Sructure

This document defines the YANG module "ietf-network-vpn-pm", which has the tree structure described in the following sub-sections.

6.1. Network Level

```
module: ietf-network-vpn-pm
  augment /nw:networks/nw:network/nw:network-types:
    +--rw network-technology-type*   identityref
  augment /nw:networks/nw:network:
    +--rw vpn-attributes
    |  +--rw vpn-topo?                  identityref
    +--rw vpn-summary-statistics
    |  +--rw ipv4
    |  |  +--rw total-routes?              uint32
    |  |  +--rw total-active-routes?       uint32
    |  +--rw ipv6
    |     +--rw total-routes?              uint32
    |     +--rw total-active-routes?       uint32
```

Figure 3: Network Level View of the hierarchies

For VPN service performance monitoring, this model defines only the following minimal set of Network level network topology attributes:

- "network-technology-type": Indicates the network technology type such as L3VPN, L2VPN, ISIS, or OSPF. If the "network-technology-type" is "VPN type" (e.g.,L3VPN, L2VPN), the "vpn-topo" MUST be set.

- "vpn-topo": The type of VPN service topology, this model supports "any-to-any", "Hub and Spoke" (where Hubs can exchange traffic), and "Hub and Spoke disjoint" (where Hubs cannot exchange traffic).

- "vpn-summary-statistics": VPN summary statistics, IPv4 statistics, and IPv6 statistics have been specified separately.

For network performance monitoring, the attributes of "Network Level" that defined in [RFC8345] do not need to be extended.

6.2. Node Level

```
augment /nw:networks/nw:network/nw:node:
    +--rw node-attributes
    |  +--rw node-type?   identityref
    |  +--rw site-id?     string
    |  +--rw site-role?   Identityref
```

Figure 4: Node Level View of the hierarchies

The Network and VPN service performance monitoring model defines only the following minimal set of Node level network topology attributes and constraints:

- "node-type" (Attribute): Indicates the type of the node, such as PE or ASBR. This "node-type" can be used to report performance metric between any two nodes each with specific node-type.

- "site-id" (Constraint): Uniquely identifies the site within the overall network infrastructure.

- "site-role" (Constraint): Defines the role of the site in a particular VPN topology.

6.3. Link and Termination Point Level

```
augment /nw:networks/nw:network/nt:link:
  +--rw link-type?                          identityref
  +--rw low-percentile                      percentile
  +--rw high-percentile                     percentile
  +--rw middle-percentile                   percentile
  +--ro reference-time                      yang:date-and-time
  +--ro measurement-interval                uint32
  +--ro link-telemetry-attributes
     +--ro loss-statistics
     |  +--ro packet-loss-count?            uint32
     |  +--ro loss-ratio?                   percentage
     |  +--ro packet-reorder-count?         uint32
     |  +--ro packets-out-of-seq-count?     uint32
     |  +--ro packets-dup-count?            uint32
     +--ro delay-statistics
     |  +--ro direction?                   identityref
     |  +--ro unit-value                   identityref
     |  +--ro min-delay-value?             yang:gauge64
     |  +--ro max-delay-value?             yang:gauge64
     |  +--ro high-delay-percentile?       yang:gauge64
     |  +--ro middle-delay-percentile?     yang:gauge64
     |  +--ro low-delay-percentile?        yang:gauge64
     +--ro jitter-statistics
        +--ro unit-value                   identityref
        +--ro min-jitter-value?            yang:gauge64
        +--ro max-jitter-value?            yang:gauge64
        +--ro low-jitter-percentile?       yang:gauge64
        +--ro high-jitter-percentile?      yang:gauge64
        +--ro middle-jitter-percentile?    yang:gauge64
augment /nw:networks/nw:network/nw:node/nt:termination-point:
  +--ro tp-telemetry-attributes
     +--ro in-octets?           uint32
     +--ro out-octets?          uint32
     +--ro inbound-unicast?     uint32
     +--ro inbound-nunicast?    uint32
     +--ro inbound-discards?    uint32
     +--ro inbound-errors?      uint32
     +--ro in-unknown-protocol?     uint32
     +--ro outbound-unicast?    uint32
     +--ro outbound-nunicast?   uint32
     +--ro outbound-discards?   uint32
     +--ro outbound-errors?     uint32
     +--ro outbound-qlen?       uint32
```

Figure 5: Link and Termination point Level View of the hierarchies

The Network and VPN service performance monitoring model defines only the following minimal set of Link level network topology attributes:

- "link-type" (Attribute): Indicates the type of the link, such as GRE or IP-in-IP.

- "low-percentile": Indicates low percentile to report. Setting low-percentile into 0.00 indicates the client is not intererested in receiving low percentile.

- "middle-percentile": Indicates middle percentile to report. Setting middle-percentile into 0.00 indicates the client is not intererested in receiving middle percentile.

- "high-percentile": Indicates high percentile to report. Setting low-percentile into 0.00 indicates the client is not intererested in receiving high percentile.

- Loss Statistics: A set of loss statistics attributes that are used to measure end to end loss between VPN sites or between any two network nodes.

- Delay Statistics: A set of delay statistics attributes that are used to measure end to end latency between VPN sites or between any two network nodes..

- Jitter Statistics: A set of IP Packet Delay Variation [RFC3393] statistics attributes that are used to measure end to end jitter between VPN sites or between any two network nodes..

The Network and VPN service performance monitoring defines the following minimal set of Termination point level network topology attributes:

- Inbound statistics: A set of inbound statistics attributes that are used to measure the inbound statistics of the termination point, such as "the total number of octets received on the termination point", "The number of inbound packets which were chosen to be discarded", "The number of inbound packets that contained errors", etc.

- Outbound statistics: A set of outbound statistics attributes that are used to measure the outbound statistics of the termination point, such as "the total number of octets transmitted out of the termination point", "The number of outbound packets which were chosen to be discarded", "The number of outbound packets that contained errors", etc.

7. Example of I2RS Pub/Sub Retrieval

This example shows the way for a client to subscribe for the Performance monitoring information between node A and node B in the L3 network topology built on top of the underlying network . The performance monitoring parameter that the client is interested in is end to end loss attribute.

```
 <rpc netconf:message-id="101"
    xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <establish-subscription
       xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
       <stream-subtree-filter>
          <networks xmlns="urn:ietf:params:xml:ns:yang:ietf-network-topo">
             <network>
              <network-id>l3-network</network-id>
              <network-technology-type xmlns="urn:ietf:params:xml:ns:yang:ietf
-network-vpn-pm">
              L3VPN
              </network-technology-type>
               <node>
                <node-id>A</node-id>
                <node-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-netwo
rk-vpn-pm">
                 <node-type>pe</node-type>
                </node-attribtues>
                <termination-point xmlns="urn:ietf:params:xml:ns:yang:ietf-net
work-topology">
                 <tp-id>1-0-1</tp-id>
                <tp-telemetry-attributes xmlns="urn:ietf:params:xml:ns:yang:ie
tf-network-vpn-pm">
                  <in-octets>100</in-octets>
                  <out-octets>150</out-octets>
                 </tp-telemetry-attributes>
                </termination-point>
               </node>
               <node>
                <node-id>B</node-id>
                <node-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-netwo
rk-vpn-pm">
                 <node-type>pe</node-type>
                </node-attribtues>
                <termination-point xmlns="urn:ietf:params:xml:ns:yang:ietf-net
work-topology">
                 <tp-id>2-0-1</tp-id>
                <tp-telemetry-attributes xmlns="urn:ietf:params:xml:ns:yang:ie
tf-network-vpn-pm">
                  <in-octets>150</in-octets>
                  <out-octets>100</out-octets>
                 </tp-telemetry-attributes>
                </termination-point>
               </node>
               <link xmlns="urn:ietf:params:xml:ns:yang:ietf-network-topology"
>
                <link-id>A-B</link-id>
                <source>
```

```
                    <source-node>A</source-node>
                   </source>
                   <destination>
                    <dest-node>B</dest-node>
                   </destination>
                    <link-type>mpls-te</link-type>
                    <link-telemetry-attributes
                     xmlns="urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm">
                     <loss-statistics>
                      <packet-loss-count>100</packet-loss-count>
                     </loss-statistics>
                    </link-telemetry-attributes>
                   </link>
                 </network>
              </networks>
           </stream-subtree-filter>
           <period xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push:1.0">500</per
iod>
       </establish-subscription>
    </rpc>
```

8. Example of RPC-based Retrieval

This example shows the way for the client to use RPC model to fetch performance data on demand, e.g., the client requests "packet-loss-count" between PE1 in site 1 and PE2 in site 2 belonging to the same VPN1.

```
    <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
        message-id="1">
     <report xmlns="urn:ietf:params:xml:ns:yang:example-service-pm-report">
      <networks xmlns="urn:ietf:params:xml:ns:yang:ietf-network-topo">
        <network>
         <network-id>vpn1</network-id>
         <node>
          <node-id>A</node-id>
          <node-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm
">
          <node-type>pe</node-type>
          </node-attribtues>
          <termination-point xmlns="urn:ietf:params:xml:ns:yang:ietf-network-topo
logy">
           <tp-id>1-0-1</tp-id>
           <tp-telemetry-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-netwo
rk-vpn-pm">
            <in-octets>100</in-octets>
            <out-octets>150</out-octets>
           </tp-telemetry-attributes>
          </termination-point>
         </node>
         <node>
          <node-id>B</node-id>
```

```
          <node-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm
">
          <node-type>pe</node-type>
          </node-attribtues>
          <termination-point xmlns="urn:ietf:params:xml:ns:yang:ietf-network-topo
logy">
           <tp-id>2-0-1</tp-id>
           <tp-telemetry-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-netwo
rk-vpn-pm">
            <in-octets>150</in-octets>
            <out-octets>100</out-octets>
           </tp-telemetry-attributes>
          </termination-point>
         </node>
         <link-id>A-B</link-id>
          <source>
           <source-node>A</source-node>
          </source>
          <destination>
           <dest-node>B</dest-node>
          </destination>
           <link-type>mpls-te</link-type>
          <telemetry-attributes xmlns="urn:ietf:params:xml:ns:yang:ietf-network-p
m">
           <loss-statistics>
            <packet-loss-count>120</packet-loss-count>
           </loss-statistics>
          </telemetry-attributes>
         </link>
       </network>
     </report>
   </rpc>
```

9. Network and VPN Service Assurance YANG Module

This module uses types defined in [RFC8345], [RFC8299] and [RFC8532].

```
<CODE BEGINS> file "ietf-network-vpn-pm@2020-04-17.yang"
module ietf-network-vpn-pm {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm";
  prefix nvp;

  import ietf-yang-types {
        prefix yang;
   reference "RFC 6991: Common YANG Types.";
  }
  import ietf-network {
    prefix nw;
    reference
      "Section 6.1 of RFC 8345: A YANG Data Model for Network
       Topologies";
```

```
  }
  import ietf-network-topology {
    prefix nt;
    reference
      "Section 6.2 of RFC 8345: A YANG Data Model for Network
       Topologies";
  }
  import ietf-l3vpn-svc {
    prefix l3vpn-svc;
    reference
      "RFC 8299: YANG Data Model for L3VPN Service Delivery";
  }
  import ietf-lime-time-types {
    prefix lime;
    reference
      "RFC 8532: Generic YANG Data Model for the Management of
       Operations, Administration, and Maintenance (OAM) Protocols
       That Use Connectionless Communications";
  }
  organization
    "IETF BESS Working Group";
  contact
    "Editor: Qin Wu
             <bill.wu@huawei.com>
     Editor: Mohamed Boucadair
             <mohamed.boucadair@orange.com>";
  description
    "This module defines a model for the VPN Service Performance
     monitoring.

     Copyright (c) 2020 IETF Trust and the persons identified as
     authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject
     to the license terms contained in, the Simplified BSD License
     set forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (http://trustee.ietf.org/license-info).

     This version of this YANG module is part of RFC XXXX; see
     the RFC itself for full legal notices.";

  revision 2019-04-17 {
    description
      "Initial revision.";
    reference
      "RFC XXXX: A YANG Model for Network and VPN Service Performance
```

```
                Monitoring";
  }

  identity network-type {
    description
      "Base type for Overlay network topology.";
  }

  identity l3vpn {
    base network-type;
    description
      "Identity for layer3 VPN network type.";
  }

  identity l2vpn {
    base network-type;
    description
      "Identity for layer2 VPN network type.";
  }

  identity ospf {
    base network-type;
    description
      "Identity for OSPF network type.";
  }

  identity isis {
    base network-type;
    description
      "Identity for ISIS network type.";
  }
  identity node-type {
    description
      "Base identity for node type";
  }

  identity pe {
    base node-type;
    description
      "Identity for PE type";
  }

  identity ce {
    base node-type;
    description
      "Identity for CE type";
  }
```

```
  identity asbr {
    base node-type;
    description
      "Identity for ASBR type";
  }

  identity p {
    base node-type;
    description
      "Identity for P type";
  }

  identity link-type {
    description
      "Base identity for link type, e.g.,GRE, MPLS TE, VXLAN.";
  }
  identity gre {
    base link-type;
    description
      "Base identity for GRE Tunnel.";
  }
  identity VXLAN {
    base link-type;
    description
      "Base identity for VXLAN Tunnel.";
  }
  identity ip-in-ip {
    base link-type;
    description
      "Base identity for IP in IP Tunnel.";
  }
  identity direction {
    description
      "Base Identity for measurement direction including
       one way measurement and two way measurement.";
  }

  identity one-way {
    base direction;
    description
      "Identity for one way measurement.";
  }

  identity two-way {
    base direction;
    description
      "Identity for two way measurement.";
  }
```

```
  typedef percentage {
    type decimal64 {
      fraction-digits 5;
      range "0..100";
    }
    description
      "Percentage.";
  }
  typedef percentile {
    type decimal64 {
      fraction-digits 2;
    }
    description
      "The nth percentile of a set of data is the
       value at which n percent of the data is below it.";
  }
     grouping vpn-summary-statistics {
       description
         "VPN Statistics grouping used for network topology
          augmentation.";
       container vpn-summary-statistics {
         description "Container for VPN summary statistics.";
         container ipv4 {
         leaf total-routes {
           type uint32;
           description
             "Total routes in the RIB from all protocols.";
         }
         leaf total-active-routes {
           type uint32;
           description
             "Total active routes in the RIB.";
         }
          description
           "IPv4-specific parameters.";
   }
    container ipv6 {
         leaf total-routes {
           type uint32;
           description
             "Total routes in the RIB from all protocols.";
         }
         leaf total-active-routes {
           type uint32;
           description
             "Total active routes in the RIB.";
         }
          description
```

```
          "IPv6-specific parameters.";
        }
    }
}

  grouping link-error-statistics {
    description
      "Grouping for per link error statistics.";
    container loss-statistics {
      description
        "Per link loss statistics.";

      leaf packet-loss-count {
        type uint32 {
          range "0..4294967295";
        }
        default "0";
        description
          "Total received packet drops count.
           The value of count will be set to zero (0)
           on creation and will thereafter increase
           monotonically until it reaches a maximum value
           of 2^32-1 (4294967295 decimal), when it wraps
           around and starts increasing again from zero.";
      }
      leaf loss-ratio {
        type percentage;
        description
          "Loss ratio of the packets. Express as percentage
           of packets lost with respect to packets sent.";
      }
      leaf packet-reorder-count {
        type uint32 {
          range "0..4294967295";
        }
        default "0";
        description
          "Total received packet reordered count.
           The value of count will be set to zero (0)
           on creation and will thereafter increase
           monotonically until it reaches a maximum value
           of 2^32-1 (4294967295 decimal), when it wraps
           around and starts increasing again from zero.";
      }
      leaf packets-out-of-seq-count {
        type uint32 {
          range "0..4294967295";
        }
```

```
        description
          "Total received out of sequence count.
           The value of count will be set to zero (0)
           on creation and will thereafter increase
           monotonically until it reaches a maximum value
           of 2^32-1 (4294967295 decimal), when it wraps
           around and starts increasing again from zero..";
      }
      leaf packets-dup-count {
        type uint32 {
          range "0..4294967295";
        }
        description
          "Total received packet duplicates count.
           The value of count will be set to zero (0)
           on creation and will thereafter increase
           monotonically until it reaches a maximum value
           of 2^32-1 (4294967295 decimal), when it wraps
           around and starts increasing again from zero.";
      }
    }
  }

  grouping link-delay-statistics {
    description
      "Grouping for per link delay statistics";
    container delay-statistics {
      description
        "Link delay summarised information. By default,
         one way measurement protocol (e.g., OWAMP) is used
         to measure delay.";
      leaf direction {
        type identityref {
          base direction;
        }
        default "one-way";
        description
          "Define measurement direction including one way
           measurement and two way measurement.";
      }
      leaf unit-value {
        type identityref {
          base lime:time-unit-type;
        }
        default "lime:milliseconds";
        description
          "Time units, where the options are s, ms, ns, etc.";
      }
```

```
      leaf min-delay-value {
        type yang:gauge64;
        description
          "Minimum delay value observed.";
      }
      leaf max-delay-value {
        type yang:gauge64;
        description
          "Maximum delay value observed.";
      }
      leaf low-delay-percentile {
        type yang:gauge64;
        description
          "Low percentile of the delay observed with
           specific measurement method.";
      }
      leaf middle-delay-percentile {
        type yang:gauge64;
        description
          "Middle percentile of the delay observed with
           specific measurement method.";
      }
      leaf high-delay-percentile {
        type yang:gauge64;
        description
          "High percentile of the delay observed with
           specific measurement method.";
      }
    }
  }

  grouping link-jitter-statistics {
    description
      "Grouping for per link jitter statistics";
    container jitter-statistics {
      description
        "Link jitter summarised information. By default,
         jitter is measured using IP Packet Delay Variation
         (IPDV).";

      leaf unit-value {
        type identityref {
          base lime:time-unit-type;
        }
        default "lime:milliseconds";
        description
          "Time units, where the options are s, ms, ns, etc.";
      }
```

```
      leaf min-jitter-value {
        type yang:gauge64;
        description
          "Minimum jitter value observed.";
      }
      leaf max-jitter-value {
        type yang:gauge64;
        description
          "Maximum jitter value observed.";
      }
      leaf low-jitter-percentile {
        type yang:gauge64;
        description
          "Low percentile of the jitter observed.";
      }
      leaf middle-jitter-percentile {
        type yang:gauge64;
        description
          "Middle percentile of the jitter observed.";
      }
      leaf high-jitter-percentile {
        type yang:gauge64;
        description
          "High percentile of the jitter observed.";
      }
    }
  }

  grouping tp-svc-telemetry {
    leaf in-octets {
      type uint32;
      description
        "The total number of octets received on the
         interface, including framing characters.";
    }
    leaf inbound-unicast {
      type uint32;
      description
        "Inbound unicast packets were received, and delivered
         to a higher layer during the last period.";
    }
    leaf inbound-nunicast {
      type uint32;
      description
        "The number of non-unicast (i.e., subnetwork-
         broadcast or subnetwork-multicast) packets
         delivered to a higher-layer protocol.";
    }
```

```
    leaf inbound-discards {
      type uint32;
      description
        "The number of inbound packets which were chosen
         to be discarded even though no errors had been
         detected to prevent their being deliverable to a
         higher-layer protocol.";
    }
    leaf inbound-errors {
      type uint32;
      description
        "The number of inbound packets that contained
         errors preventing them from being deliverable to a
         higher-layer protocol.";
    }
    leaf outbound-errors {
      type uint32;
      description
        "The number of outbound packets that contained
         errors preventing them from being deliverable to a
         higher-layer protocol.";
    }
    leaf in-unknown-protocol {
      type uint32;
      description
        "The number of packets received via the interface
         which were discarded because of an unknown or
         unsupported protocol.";
    }
    leaf out-octets {
      type uint32;
      description
        "The total number of octets transmitted out of the
         interface, including framing characters.";
    }
    leaf outbound-unicast {
      type uint32;
      description
        "The total number of packets that higher-level
         protocols requested be transmitted to a
         subnetwork-unicast address, including those that
         were discarded or not sent.";
    }
    leaf outbound-nunicast {
      type uint32;
      description
        "The total number of packets that higher-level
         protocols requested be transmitted to a non-
```

```
          unicast (i.e., a subnetwork-broadcast or
          subnetwork-multicast) address, including those
          that were discarded or not sent.";
      }
      leaf outbound-discards {
        type uint32;
        description
          "The number of outbound packets which were chosen
           to be discarded even though no errors had been
           detected to prevent their being transmitted.  One
           possible reason for discarding such a packet could
           be to free up buffer space.";
      }
      leaf outbound-qlen {
        type uint32;
        description
          " Length of the queue of the interface from where
            the packet is forwarded out.  The queue depth could
             be the current number of memory buffers used by the
            queue and a packet can consume one or more memory buffers
            thus constituting device-level information.";
      }
      description
        "Grouping for interface service telemetry.";
    }

    augment "/nw:networks/nw:network/nw:network-types" {
      description
        "Augment the network-types with service topologyies types";
      leaf-list network-technology-type {
        type identityref {
          base network-type;
        }
        description
          "Identify the network technology type, e.g., L3VPN,
           L2VPN, ISIS, OSPF.";
      }
    }
    augment "/nw:networks/nw:network" {
      description
        "Augment the network with service topology attributes";
      container vpn-topo-attributes {
        leaf vpn-topology {
          type identityref {
            base l3vpn-svc:vpn-topology;
          }
          description
            "VPN service topology, e.g., hub-spoke, any-to-any,
```

```
            hub-spoke-disjoint";
        }
        description
          "Container for vpn topology attributes.";
      }
      uses vpn-summary-statistics;
    }
    augment "/nw:networks/nw:network/nw:node" {
      description
        "Augment the network node with overlay topology attributes";
      container node-attributes {
        leaf node-type {
          type identityref {
            base node-type;
          }
          description
            "Node type, e.g., PE, P, ASBR.";
        }
        leaf site-id {
          type string;
          description
            "Associated vpn site";
        }
        leaf site-role {
          type identityref {
            base l3vpn-svc:site-role;
          }
          default "l3vpn-svc:any-to-any-role";
          description
            "Role of the site in the VPN.";
        }
        description
          "Container for overlay topology attributes.";
      }
    }
    augment "/nw:networks/nw:network/nt:link" {
      description
        "Augment the network topology link with overlay topology attributes";
       leaf link-type {
          type identityref {
            base link-type;
          }
          description
            "Link type, e.g., GRE,VXLAN, IP in IP.";
        }
       leaf low-percentile {
         type percentile;
         default 10.00;
```

```
       description
         "Low percentile to report.Setting low-percentile into 0.00 indicates
          the client is not intererested in receiving low percentile.";
     }
     leaf middle-percentile {
       type percentile;
       default 50.00;
       description
         "Middle percentile to report.Setting middle-percentile into 0.00 indicat
es
         the client is not intererested in receiving middle percentile.";
     }
     leaf high-percentile {
       type percentile;
       default 90.00;
       description
         "High percentile to report.";
     }
      leaf reference-time {
       type yang:date-and-time;
       description
         "The time that the current Measurement Interval started.Setting high-per
centile
         into 0.00 indicates the client is not intererested in receiving high per
centile.";
       }
     leaf measurement-interval {
       type uint32;
           units "seconds";
           default 60;
           description
             "Interval to calculate performance metric.";
     }
    container link-telemetry-attributes {
      config false;
      uses link-error-statistics;
      uses link-delay-statistics;
      uses link-jitter-statistics;
      description
        "Container for service telemetry attributes.";
    }
  }
  augment "/nw:networks/nw:network/nw:node/nt:termination-point" {
    description
      "Augment the network topology termination point with vpn service attributes
";
    container tp-telemetry-attributes {
      config false;
      uses tp-svc-telemetry;
      description
        "Container for termination point service telemetry attributes.";
    }
```

```
  }
}
<CODE ENDS>
```

10. Security Considerations

The YANG modules defined in this document MAY be accessed via the RESTCONF protocol [RFC8040] or NETCONF protocol ([RFC6241]). The lowest RESTCONF or NETCONF layer requires that the transport-layer protocol provides both data integrity and confidentiality, see Section 2 in [RFC8040] and [RFC6241]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH)[RFC6242] . The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC5246].

The NETCONF access control model [RFC6536] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

- /nw:networks/nw:network/svc-topo:svc-telemetry-attributes

- /nw:networks/nw:network/nw:node/svc-topo:node-attributes

11. IANA Considerations

This document requests IANA to register the following URI in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

```
URI: urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.
```

This document requests IANA to register the following YANG module in the "YANG Module Names" subregistry [RFC6020] within the "YANG Parameters" registry.

```
Name:         ietf-network-vpn-pm
Namespace:    urn:ietf:params:xml:ns:yang:ietf-network-vpn-pm
Maintained by IANA: N
Prefix:       nvp
Reference:    RFC XXXX
```

12. Contributors

Michale Wang
Huawei
Email:wangzitao@huawei.com

Roni Even
Huawei
Email: ron.even.tlv@gmail.com

13. References

13.1. Normative References

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

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, DOI 10.17487/RFC3393, November 2002, <https://www.rfc-editor.org/info/rfc3393>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6370] Bocci, M., Swallow, G., and E. Gray, "MPLS Transport Profile (MPLS-TP) Identifiers", RFC 6370, DOI 10.17487/RFC6370, September 2011, <https://www.rfc-editor.org/info/rfc6370>.

[RFC6374] Frost, D. and S. Bryant, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, DOI 10.17487/RFC6374, September 2011, <https://www.rfc-editor.org/info/rfc6374>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <https://www.rfc-editor.org/info/rfc6536>.

[RFC7923] Voit, E., Clemm, A., and A. Gonzalez Prieto, "Requirements for Subscription to YANG Datastores", RFC 7923, DOI 10.17487/RFC7923, June 2016, <https://www.rfc-editor.org/info/rfc7923>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC7952] Lhotka, L., "Defining and Using Metadata with YANG", RFC 7952, DOI 10.17487/RFC7952, August 2016, <https://www.rfc-editor.org/info/rfc7952>.

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

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8532] Kumar, D., Wang, Z., Wu, Q., Ed., Rahman, R., and S. Raghavan, "Generic YANG Data Model for the Management of Operations, Administration, and Maintenance (OAM) Protocols That Use Connectionless Communications", RFC 8532, DOI 10.17487/RFC8532, April 2019, <https://www.rfc-editor.org/info/rfc8532>.

13.2. Informative References

[I-D.ietf-netconf-yang-push]
Clemm, A. and E. Voit, "Subscription to YANG Datastores", draft-ietf-netconf-yang-push-25 (work in progress), May 2019.

[RFC4176] El Mghazli, Y., Ed., Nadeau, T., Boucadair, M., Chan, K., and A. Gonguet, "Framework for Layer 3 Virtual Private Networks (L3VPN) Operations and Management", RFC 4176, DOI 10.17487/RFC4176, October 2005, <https://www.rfc-editor.org/info/rfc4176>.

[RFC7471] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <https://www.rfc-editor.org/info/rfc7471>.

[RFC7810] Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 7810, DOI 10.17487/RFC7810, May 2016, <https://www.rfc-editor.org/info/rfc7810>.

[RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <https://www.rfc-editor.org/info/rfc8233>.

[RFC8299] Wu, Q., Ed., Litkowski, S., Tomotaki, L., and K. Ogaki, "YANG Data Model for L3VPN Service Delivery", RFC 8299, DOI 10.17487/RFC8299, January 2018, <https://www.rfc-editor.org/info/rfc8299>.

[RFC8571] Ginsberg, L., Ed., Previdi, S., Wu, Q., Tantsura, J., and C. Filsfils, "BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions", RFC 8571, DOI 10.17487/RFC8571, March 2019, <https://www.rfc-editor.org/info/rfc8571>.

Authors' Addresses

Qin Wu (editor)
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Mohamed Boucadair (editor)
Orange
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Oscar Gonzalez de Dios
Telefonica
Madrid
ES

Email: oscar.gonzalezdedios@telefonica.com

Bin Wen
Comcast

Email: bin_wen@comcast.com

Change Liu
China Unicom

Email: liuc131@chinaunicom.cn

Honglei Xu
China Telecom

Email: xuhl.bri@chinatelecom.cn