Network Working Group
Internet-Draft
Intended status: Proposed Standard
Expires: November 2, 2016

A. Verma
Juniper Networks

J. Drake
Juniper Networks

R. Molina
Ericsson Inc.

W. Lin
Juniper Networks

May 2, 2016

Vpls Best-site id
draft-anshuverma-bess-vpls-best-site-id-02.txt

Abstract

With network-based applications becoming prevalent, solutions that provide connectivity over wide area become more attractive for customers. In small-to-medium enterprise sector, Virtual Private LAN Service (VPLS), is a very useful service provider offering. It creates an emulated LAN segments fully capable of learning and forwarding Ethernet MAC addresses.

Today, in VPLS implementations, within the context of a VPLS PE (VE), a single-site is selected from which all PWs are rooted. The site-election mechanism is usually hard-coded by different vendors (e.g. minimum or maximum site-id), and as such, is outside end-users control. This offers no flexibility to end-users as it forces them to define the site-id allocation scheme well in advance, or deal with the consequences of a suboptimal site-id election. Moreover, whenever the elected site-id is declared down, the traffic to and from all other sites hosted within the same VE is impacted as well.

This draft defines protocol extensions to keep core-facing pseudowires (PWs) established at all times, regardless of the events

taking place on the attachment-circuit (AC) segment when using the BGP-based signaling 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.

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

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

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

This Internet-Draft will expire on November 2, 2016.

Copyright Notice

Table of Contents

1. Introduction

As the popularity of VPLS services continue to expand, Service Provider requirements for a scalable multi-homed solution are becoming increasingly demanding. As dictated by RFC4762 BGP-VPLS RFC, every PE participating in a VPLS domain must be fully meshed through a bidirectional pseudowire (PW). This set of PWs is built attending to the signaling information (label-block) advertised by each PE. The label-block used to build any given PW, will be the one matching the local site being elected as ’representative’ of the VPLS domain within a given PE. As stated in RFC4762, if this site is ever declared ’down’, a compliant implementation will need to either withdraw the corresponding label-block, or announce that the affected site is no longer reachable. In either case, the PW will end up being destroyed, which will have a considerable impact on other local sites relying on this specific PW. Furthermore, as a considerable amount of cycles are spent in destroying/re-building affected PWs, the overall convergence period will be severely impacted for those critical multi-homed sites that need a rapid transition to a backup PE.

This draft defines protocol extensions to keep core-facing pseudowires established at all times, regardless of the events taking place on the attachment-circuit segment when using the BGP-based signaling procedures defined in [RFC4761].

Today, in VPLS implementations, within the context of a VPLS_PE (VE), a single-site is selected from which all PWs are rooted. The site-election mechanism is usually hard-coded by different vendors (e.g. minimum or maximum site-id), and as such, is outside end-users control. This offers no flexibility to end-users as it forces them to define the site-id allocation scheme well in advance, or deal with the consequences of a suboptimal site-id election. Moreover, whenever the elected site-id is declared down, the traffic to and from all other sites hosted within the same VE is impacted as well.

In BGP VPLS MH scenarios the above pitfalls are specially acute, as not only we need to factor in the cost to bring the active PW down and run DF election in primary PE, but also in the n-DF PE and all remote-PEs within the VPLS domain. Taking into account that control-plane operation is signaled through BGP protocol, is fare to expect that many of these operations will be carried out in sequence and not in parallel, so the overall cost is usually pretty considerable in scaling scenarios.

To achieve minimal traffic disruption, this draft introduces a virtual or dummy site which will serve as the preferable or best site within each VE. Thereby, its corresponding site-id value will be defined by the end-user. But more than providing greater provisioning flexibility, the real advantage of this best-site solution relies on the capability to maintain VPLS PWs established at all times regardless of the fluctuations in AC segments.

To summarize, this best-site feature offers:

* Greater provisioning flexibility.
* Minimal traffic disruption for non-preferable sites in multi-site VEs (upon AC going down).
* Convergence period would be considerably reduced in MH setups during transient intervals.

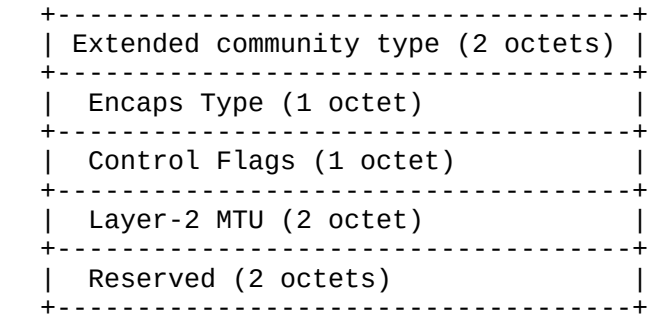
2. Conventions used in this document

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

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

3. Modifications to Layer 2 Info Extended Community

The Layer 2 Info Extended Community is used to signal control information of the pseudowires to be setup. The extended community format is described in [RFC4761]. This draft recommends that the Control Flags field of this extended community be used to synchronize the best-site information amongst PEs for a given L2VPN.



Layer-2 Info Extended Community:

Control Flags Bit Vector:

This field contains bit flags relating to pseudowire’s control information. It is augmented with the definition of one new flag field. If on a given PE VPLS instance is configured with ’best-site’, it will include in its VPLS BGP NLRI a Layer 2 Info Extended Community using Control Flags field with B = 1.

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|D|A|F|B|T|R|C|S| (Z = MUST Be Zero)
+-+-+-+-+-+-+-+-+
```

With reference to the Control Flags Bit Vector, the following bits in the Control Flags are defined; the remaining bits, MUST be set to zero when sending and MUST be ignored when receiving this Extended Community. The signaling procedure described here is therefore backwards compatible with existing implementations.

D Defined in l2vpn-vpls-multihoming draft

A Defined in l2vpn-auto-site-id draft

F Defined in l2vpn-vpls-multihoming draft

B When the bit value is 1, the PE receiving the label-block will deem the corresponding site as the most preferable site from the remote neighbor.
When the bit value is 0, the PE receiving the label-block will rely on its legacy/default site-election algorithm.

T/R Defined in l2vpn-fat-pw-bgp draft

C Defined in [RFC4761]

S Defined in [RFC4761]

4. Best-site functionality:

Traditionally, vpls path selection mechanism pick the minimum (or maximum) site-id to determine the ’preferable’ local site. This ’preferable’ local site serves two purposes: 1) pseudowires created from the local VE will be rooted from this site, and 2) pseudowires created from remote VEs will be built towards this elected site.

In order to provide some greater flexibility in the current pre-defined site-election process, this draft proposes a solution to give priority to these ’best-sites’ in detriment of those local sites with minimum (maximum) site-ids.

This solution would be fully backward compatible as VPLS-PEs on which the proposed feature isn’t enable, would simply obviate the BGP extensions previously described, and thereby, would rely on their legacy/default site-election mechanism.

Let’s make use of the following example to describe our solution in more details:

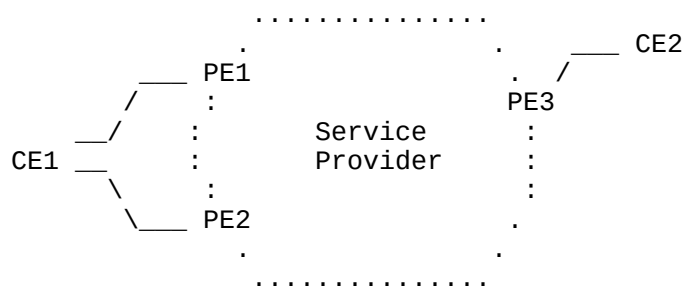


Figure 1- MH scenario with Best-site capable nodes.

A PE where ’best-site’ feature is enabled in VPLS instance, behaves as a dummy site and no access interface will be associated with it. This dummy site won’t be subjected to access interface down/up events; thereby, the corresponding D-bit will not be set to represent a site-down condition. The main goal here is to have a site that is permanently alive, regardless of the state of the attached circuits defined within the VPLS domain.

Each VPLS instance where a ’best-site’ is defined (e.g. PE1), will signal the site’s existence by setting the B-bit of the control-flags bit-vector within the L2-info extended community. Upon arrival of this BGP advertisement to the receiving PE (e.g. PE3), and only if this one is ’best-site’ capable, the received B-bit will be honored and the corresponding site will be elected as the most preferable site within the remote VE (PE1).

For those neighbors where 'best-site' feature is not configured, conventional local site election will take place. For instance, if PE1 does not receive a Label-Block advertisement with B-bit set from a remote PE (PE3), it will assume that PE3 is not 'best-site' capable, and will create a pseudowire from its minimum (maximum) designated site. For the rest of the 'best-site' capable PEs, PE1 will construct pseudowires rooted at its 'best-site' site.

By proceeding to define a 'best-site' in each of the VEs across the VPLS network, we will be drastically reducing the DF transition period as no CPU cycles will need to be spent destroying and creating new pseudowires during failover events.

5. Remote mac-flush requirement:

Having a permanent pseudowire setup would not be that effective if we end up relying solely on the current implicit mac-flush mechanism. MAC addresses are automatically aged out when the pseudowire over which they are learned is deleted. This approach would collide with the proposed 'best-site' feature, in which pseudowires are kept established on a permanent basis.

An explicit-mac-flush capable implementation would ensure that MAC-to-pseudowire bindings are cleared the moment in which a DF transition is initiated. In scenarios where 'best-site' feature is enabled, no core-facing PW will be ever torn down, so previously learned MAC entries could potentially end up pointing to an invalid PW.

Thereby, to avoid potential traffic blackholes, any successful 'best-site' implementation should be capable of supporting the explicit-mac-flush mechanism depicted in [I-D.ietf-l2vpn-vpls-multihoming draft]. F-bit was introduced in the Control-Flags bit-vector, to provide a deterministic method in which any given PE can request a remote PE to flush those mac-entries learned from the former one.

Control Flags Bit Vector

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|D|A|F|B|Z|Z|C|S| (Z = MUST Be Zero)
+-+-+-+-+-+-+-+-+
```

When making use of this feature, a DF PE will set the 'F' bit, whereas an n-DF one will clear it when sending BGP MH advertisements. A state transition from one to zero for the 'F' bit, will be interpreted by a remote PE as an indication to flush all the MACs learned from the PE that is transitioning from DF to n-DF.

6. Security Considerations

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

7. IANA Considerations

8. References

8.1. Normative References

[I-D.ietf-l2vpn-vpls-multihoming] Kothari, B., Kompella, K., Henderickx, W., Balus, F., Uttaro, J., Palislamovic, S., and W. Lin, "BGP based Multi-homing in Virtual Private LAN Service", draft-ietf-l2vpn-vpls-multihoming-07, May 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

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

8.2. Informative References

[RFC4447] Martini, L., Rosen, E., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", RFC 4447, April 2006.

[RFC6624] Kompella, K., Kothari, B., and R. Cherukuri, "Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling", RFC 6624, May 2012.

9. Author's Addresses

Anshu Verma
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089, USA

Email: anshuverma@juniper.net

John Drake
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089, USA

Email: jdrake@juniper.net

Rodny Molina
Ericsson Inc.
100 Headquarters Dr,
San Jose, CA 95134

Email: rodny.molina.maldonado@ericsson.com

Wen Lin
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089, USA

Email: wlin@juniper.net

INTERNET-DRAFT
Intended Status: Informational
Expires: January 6, 2016

R. Fernando
Cisco
S. Mackie
Juniper
D. Rao
Cisco
B. Rijsman
Juniper
M. Napierala
AT&T

July 5, 2015

Service Chaining using Virtual Networks with BGP

draft-fm-bess-service-chaining-01

Abstract

This document describes how service function chains (SFC) can be applied to traffic flows using routing in a virtual (overlay) network to steer traffic between service nodes. Chains can include services running in routers, on physical appliances or in virtual machines. Service chains have applicability at the subscriber edge, business edge and in multi-tenant datacenters. The routing function into SFCs and between service functions within an SFC can be performed by physical devices (routers), be virtualized inside hypervisors, or run as part of a host OS.

A BGP control plane for route distribution is used to create virtual networks implemented using IP MPLS, VXLAN or other suitable encapsulation, where the routes within the virtual networks cause traffic to flow through a sequence of service nodes that apply packet processing functions to the flows. Two techniques are described: in one the service chain is implemented as a sequence of distinct VPNs between sets of service nodes that apply each service function; in the other, the routes within a VPN are modified through the use of special route targets and modified next-hop resolution to achieve the desired result.

In both techniques, service chains can be created by manual configuration of routes and route targets in routing systems, or through the use of a controller which contains a topological model of the desired service chains.

This document also contains discussion of load balancing between

network functions, symmetric forward and reverse paths when stateful services are involved, and use of classifiers to direct traffic into a service chain.

Status of this Memo

Table of Contents

1 Introduction

The purpose of networks is to allow computing systems to communicate with each other. Requests are usually made from the client or customer side of a network, and responses are generated by applications residing in a datacenter. Over time, the network between the client and the application has become more complex, and traffic between the client and the application is acted on by intermediate systems that apply network services. Some of these activities, like firewall filtering, subscriber attachment and network address translation are generally carried out in network devices along the traffic path, while others are carried out by dedicated appliances, such as media proxy and deep packet inspection (DPI). Deployment of these in-network services is complex, time-consuming and costly, since they require configuration of devices with vendor-specific operating systems, sometimes with co-processing cards, or deployment of physical devices in the network, which requires cabling and configuration of the devices that they connect to. Additionally, other devices in the network need to be configured to ensure that traffic is correctly steered through the systems that services are running on. The current mode of operations does not easily allow common operational processes to be applied to the lifecycle of services in the network, or for steering of traffic through them. The recent emergence of Network Functions Virtualization (NFV) [NFVE2E] to provide a standard deployment model for network services as software appliances, combined with Software Defined Networking (SDN) for more dynamic traffic steering can provide foundational elements that will allow network services to be deployed and managed far more efficiently and with more agility than is possible today. This document describes how the combination of several existing technologies can be used to create chains of functions, while preserving the requirements of scale, performance and reliability for service provider networks. The technologies employed are:

- Traffic flow between service functions described by routing and

network policies rather than by static physical or logical connectivity

o Packet header encapsulation in order to create virtual private networks using network overlays

o VRFs on both physical devices and in hypervisors to implement forwarding policies that are specific to each virtual network

o Optional use of a controller to calculate routes to be installed in routing systems to form a service chain. The controller uses a topological model that stores service function instance connectivity to network devices and intended connectivity between service functions.

o MPLS or other labeling to facilitate identification of the next interface to send packets to in a service function chain

o BGP or BGP-style signaling to distribute routes in order to create service function chains

o Distributed load balancing between service functions performed in the VRFs that service function instance connect to.

Virtualized environments can be supported without necessarily running BGP or MPLS natively. Messaging protocols such as NC/YANG, XMPP or OpenFlow may be used to signal forwarding information. Encapsulation mechanisms such as VXLAN or GRE may be used for overlay transport. The term "BGP-style", above, refers to this type of signaling.

Traffic can be directed into service function chains using IP routing at each end of the service function chain, or be directed into the chain by a classifier function that can determine which service chain a traffic flow should pass through based on deep packet inspection (DPI) and/or subscriber identity.

The techniques can support an evolution from services implemented in physical devices attached to physical forwarding systems (routers) to fully virtualized implementations as well as intermediate hybrid implementations.

1.1 Terminology

This document uses the following acronyms and terms.

Terms	Meaning
AS	Autonomous System
ASBR	Autonomous System Border Router
CE	Customer Edge
FW	Firewall
I2RS	Interface to the Routing System
L3VPN	Layer 3 VPN
LB	Load Balancer
NLRI	Network Layer Reachability Information [RFC4271]
P	Provider backbone router
proxy-arp	proxy-Address Resolution Protocol
RR	Route Reflector
RT	Route Target
SDN	Software Defined Network
vCE	virtual Customer Edge router
vFW	virtual Firewall
vLB	virtual Load Balancer
VM	Virtual Machine
vPC	virtual Private Cloud
vPE	virtual Provider Edge router
VPN	Virtual Private Network
VRF	VPN Routing and Forwarding table [RFC4364]
vRR	virtual Route Reflector

This document follows some of the terminology used in [draft-ietf-sfc-architecture] and adds some new terminology:

Network Service:
: An externally visible service offered by a network operator; a service may consist of a single service function or a composite built from several service functions executed in one or more pre-determined sequences and delivered by software executing in physical or virtual devices

Classification:
: Customer/network/service policy used to identify and select traffic flow(s) requiring certain outbound forwarding actions, in particular, to direct specific traffic flows into the ingress of a particular service function chain, or causing branching within a service function chain.

Virtual Network:

A logical overlay network built using virtual links or packet encapsulation, over an existing network (the underlay).

Service Function Chain (SFC):
A service function chain defines an ordered set of service functions that must be applied to packets and/or frames selected as a result of classification. An SFC may be either a linear chain or a complex service graph with multiple branches. The term "Service Chain" is often used in place of "Service Function Chain".

SFC Set:

The pair of SFCs through which the forward and reverse directions of a given classified flow will pass.

Service Function (SF):
A logical function that is applied to packets. A service function can act at the network layer or other OSI layers. A service function can be embedded in one or more physical network elements, or can be implemented in one or more software instances running on physical or virtual hosts. One or multiple service functions can be embedded in the same network element or run on the same host. Multiple instances of a service function can be enabled in the same administrative domain. We will also refer to "Service Function" as, simply, "Service" for simplicity.

A non-exhaustive list of services includes: firewalls, DDOS protection, anti-malware/ant-virus systems, WAN and application acceleration, Deep Packet Inspection (DPI), server load balancers, network address translation, HTTP Header Enrichment functions, video optimization, TCP optimization, etc.

SF Instance:
An instance of software that implements the packet processing of a service function

SF Instance Set:
A group of SF instances that, in parallel, implement a service function in an SFC. Routing System: A hardware or software system that performs layer 3 routing and/or forwarding functions. The term includes physical routers as well as hypervisor or Host OS implementations of the forwarding plane of a conventional router.

VRF:
A subsystem within a routing system as defined in [RFC4364] that contains private routing and forwarding tables and has physical and/or logical interfaces associated with it. In the case of

hypervisor/Host OS implementations, the term refers only to the forwarding function of a VRF, and this will be referred to as a "VPN forwarder."

Ingress VRF:
A VRF containing an ingress interface of a SF instance

Egress VRF:
A VRF containing an egress interface of a SF instance

2 Service Function Chain Architecture Using Virtual Networking

The techniques described in this document use virtual networks to implement service function chains. Service function chains can be implemented on devices that support existing MPLS VPN and BGP standards [RFC4364, RFC4271, RFC4760], but other encapsulations, such as VXLAN [RFC7348], can be used. Similarly, equivalent control plane protocols such as BGP-EVPN can also be used where supported.

The following sections detail the building blocks of the SFC architecture, and outline the processes of route installation and subsequent route exchange to create an SFC.

2.1 High Level Architecture

Service function chains can be deployed with or without a classifier. Use cases where SFCs may be deployed without a classifier include multi-tenant data centers, private and public cloud and virtual CPE for business services. Classifiers will primarily be used in mobile and wireline subscriber edge use cases. Use of a classifier is discussed in Section 4.

A high-level architecture diagram of an SFC without a classifier, where traffic is routed into and out of the SFC, is shown in Figure 1, below. An optional controller is shown that contains a topological model of the SFC and which configures the network resources to implement the SFC.

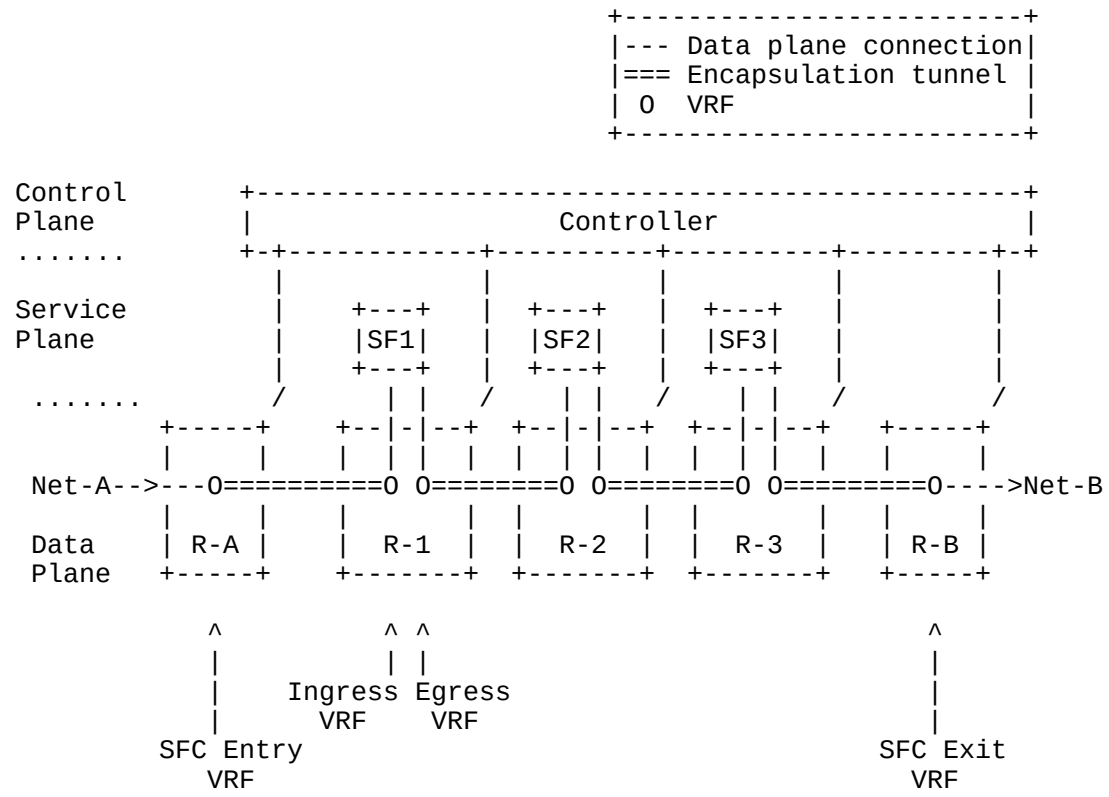


Figure 1 - High level SFC Architecture

Traffic from Network-A destined for Network-B will pass through the SFC composed of SF instances, SF1, SF2 and SF3. Routing system R-A contains a VRF (shown as "O" symbol) that is the SFC entry point. This VRF will advertise a route to reach Network-B into Network-A causing any traffic from a source in Network-A with a destination in Network-B to arrive in this VRF. The forwarding table in the VRF in R-A will direct traffic destined for Network-B into an encapsulation tunnel with destination R-1 and a label that identifies the ingress (left) interface of SF1 that R-1 should send the packets out on. The packets are processed by service instance SF-1 and arrive in the egress (right) VRF in R-1. The forwarding entries in the egress VRF direct traffic to the next ingress VRF using encapsulation tunneling. The process is repeated for each service instance in the SFC until packets arrive at the SFC exit VRF (in R B). This VRF is peered with Network-B and routes packets towards their destinations in the user data plane.

In the example, each pair of ingress and egress VRFs are configured

in separate routing systems, but such pairs could be collocated in the same routing system, and it is possible for the ingress and egress VRFs for a given SF instance to be in different routing systems. The SFC entry and exit VRFs can be collocated in the same routing system, and the service instances can be local or remote from either or both of the routing systems containing the entry and exit VRFs, and from each other.

The controller is responsible for configuring the VRFs in each routing system, installing the routes in each of the VRFs to implement the SFC, and, in the case of virtualized services, may instantiate the service instances.

2.2 Service Function Chain Logical Model

A service function chain is a set of logically connected service functions through which traffic can flow. Each egress interface of one service function is logically connected to an ingress interface of the next service function.

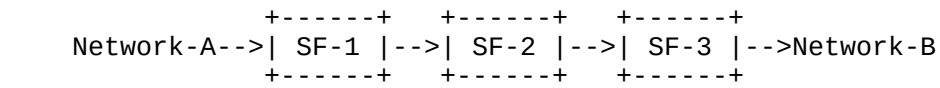


Figure 2 - A Chain of Service Functions

In Figure 2, above, a service function chain has been created that connects Network-A to Network-B, such that traffic from a host in Network-A to a host in Network-B will traverse the service function chain.

As defined in [draft-ietf-sfc-architecture], a service function chain can be uni-directional or bi-directional. In this document, in order to allow for the possibility that the forward and reverse paths may not be symmetrical, SFCs are defined as uni-directional, and the term "SFC set" is used to refer to a pair of forward and reverse direction SFCs for some set of routed or classified traffic.

2.3 Service Function Implemented in a Set of SF Instances

A service function instance is a software system that acts on packets that arrive on an ingress interface of that software system. Service function instances may run on a physical appliance or in a virtual machine. A service function instance may be transparent at layer 2 and/or 3, and may support branching across multiple egress interfaces and may support aggregation across ingress interfaces. For simplicity, the examples in this document have a single ingress and a single egress interface.

Each service function in a chain can be implemented by a single service function instance, or by a set of instances in order to provide scale and resilience.

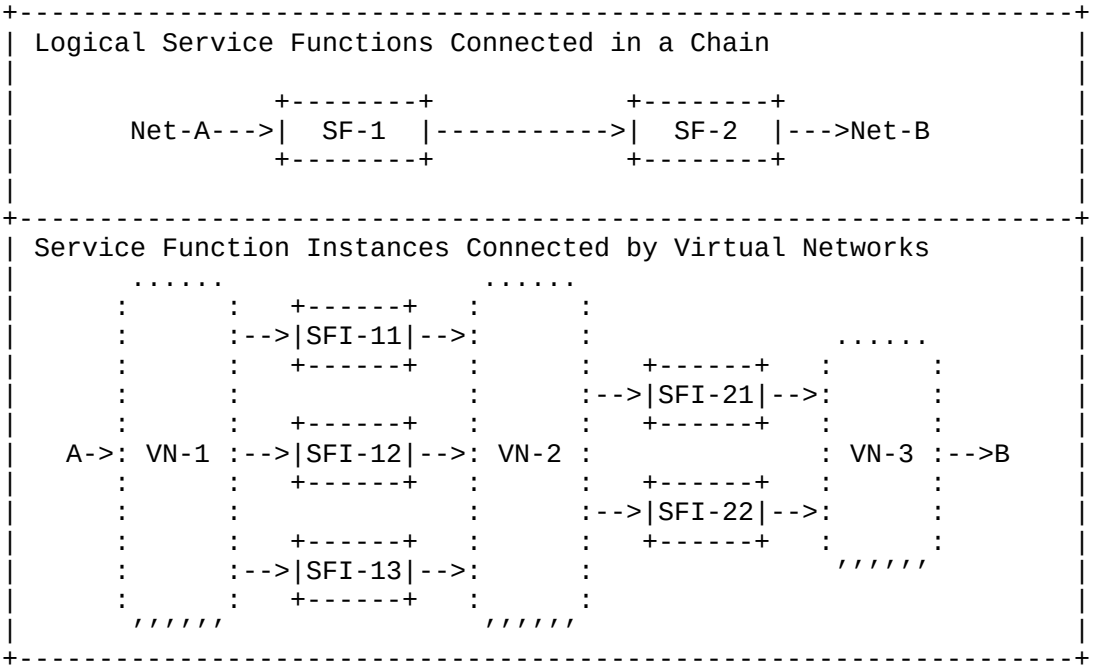


Figure 3 - Service Functions Are Composed of SF Instances Connected Via Virtual Networks

In Figure 3, service function SF-1 is implemented in three service function instances, SFI-11, SFI-12, and SFI-13. Service function SF-2 is implemented in two SF instances. The service function instances are connected to the next service function in the chain using a virtual network, VN-2. Additionally, a virtual network (VN-1) is used to enter the SFC and another (VN-3) is used at the exit.

The logical connection between two service functions is implemented using a virtual network that contains egress interfaces for instances of one service function, and ingress interfaces of instances of the next service function. Traffic is directed across the virtual network between the two sets of service function instances using layer 3 forwarding (e.g. an MPLS VPN) or layer 2 forwarding (e.g. a VXLAN).

The virtual networks could be described as "directed half-mesh", in

that the egress interface of each SF instance of one service function can reach any ingress interface of the SF instances of the connected service function.

Details on how routing across virtual networks is achieved, and requirements on load balancing across ingress interfaces are discussed in later sections of this document.

2.4 SF Instance Connections to VRFs

SF instances can be deployed as software running on physical appliances, or in virtual machines running on a hypervisor. These two options are described in more detail in the following sections.

2.4.1 SF Instance in Physical Appliance

The case of a SF instance running on a physical appliance is shown in Figure 4, below.

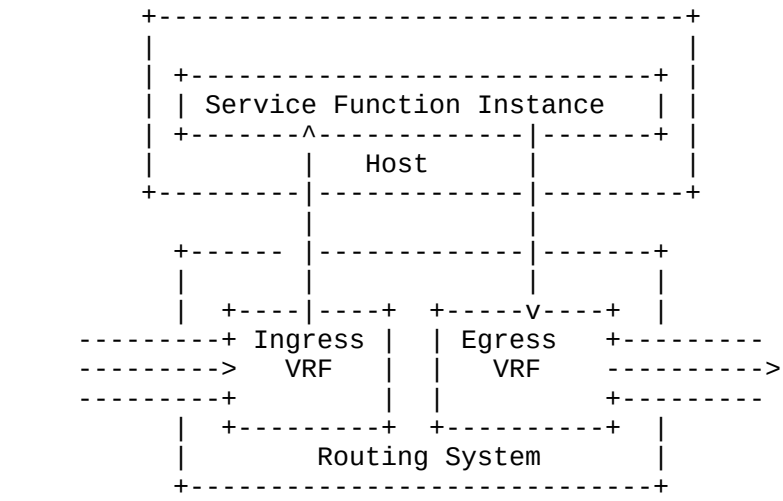


Figure 4 - Ingress and Egress VRFs for a Physical Routing System and Physical SF Instance

The routing system is a physical device and the service function instance is implemented as software running in a physical appliance (host) connected to it. Transport between VRFs on different routing systems that are connected to other SF instances in an SFC is via encapsulation tunnels, such as MPLS over GRE, or VXLAN.

2.4.2 SF Instance in a Virtualized Environment

In virtualized environments, a routing system with VRFs that act as VPN forwarders is resident in the hypervisor/Host OS, and is co-resident in the host with one or more SF instances that run in virtual machines. The egress VPN forwarder performs tunnel encapsulation to send packets to other physical or virtual routing systems with attached SF instances to form an SFC. The tunneled packets are sent through the physical interfaces of the host to the other hosts or physical routers. This is illustrated in Figure 5, below.

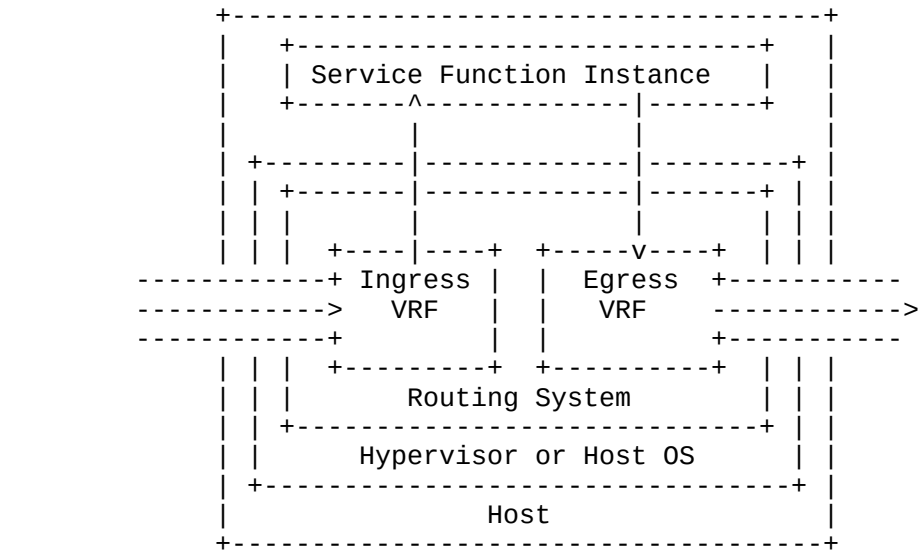


Figure 5 - Ingress and Egress VRFs for a Virtual Routing System and Virtualized SF Instance

When more than one instance of an SF is running on a hypervisor, they can be connected to the same VRF for scale out of an SF within an SFC.

The routing mechanisms in the VRFs into and between service function instances, and the encapsulation tunneling between routing systems are identical in the physical and virtual implementation of SFCs described in this document. Physical and virtual service functions can be mixed as needed with different combinations of physical and virtual routing systems.

2.5 Encapsulation Tunneling for Transport

Encapsulation tunneling is used to transport packets between SF

instances in the chain and, when a classifier is not used, from the originating network into the SFC and from the SFC into the destination network.

The tunnels can be MPLS over GRE [RFC4023], MPLS over UDP [draft-ietf-mpls-in-udp], MPLS over MPLS [RFC3031], VXLAN [RFC7348], or another suitable encapsulation method.

Tunneling may be enabled in each routing system as part of a base configuration or may be configured by the controller. Tunnel encapsulations may be configured by the controller or signaled using BGP.

2.6 SFC Creation Procedure

This section describes how service chains are created using two methods:

o Sequential VPNs - where a conventional VPN is created between each set of SF instances to create the links in the SFC

o Route Modification - where each routing system modifies advertised routes that it receives, to realize the links in an SFC on the basis of a special service topology RT and a route-policy that describes the service chain logical topology

In both cases the controller, when present, is responsible for creating ingress and egress VRFs, configuring the interfaces connected to SF instances in each VRF and configuring RTs for each VRF. Additionally, in the second method, the controller also sends the route-policy containing the service chain logical topology to each routing system. If a controller is not used, these procedures will require to be performed manually or through scripting, for instance.

The following sub-sections describe how RT configuration, local route installation and route distribution occurs in each of the methods.

2.6.1 SFC Provisioning Using Sequential VPNs

The task of the controller in this method of SFC provisioning is to create a set of VPNs that carry traffic to the destination network through instances of each service function in turn. This is achieved by configuring RTs such that the egress VRFs of one set of SF instances import an RT that is an export RT for the ingress VRFs of the next, logically connected, set of SF instances.

The process of SFC creation is as follows

1. Controller creates a VRF in each routing system that is connected to a service instance that will be used in the SFC

2. Controller configures each VRF to contain the logical interface that connects to a SF instance.

3. Controller implements route target import and export policies in the VRFs using the same route targets for the egress VRFs of a service function and the ingress VRFs of the next logically connected service function in the SFC.

4. Controller installs a static route in each ingress VRF whose next hop is the interface that a SF instance is connected to. The prefix for the route is the destination network to be reached by passing through the SFC.

5. Routing systems advertise the static routes via BGP as VPN routes with next hop being the IP address of the router, with an encapsulation specified and a label that identifies the service instance interface.

6. Routing systems containing VRFs with matching route targets receive the updates.

7. Routes are installed in egress VRFs with matching import targets. The egress VRFs of each SF instance will now contain VPN routes to one or more routers containing ingress VRFs for SF instances of the next service function in the SFC.

In the case of physical routers, the creation and configuration of VRFs, interfaces and local static routes can be performed programmatically using Netconf; and BGP route distribution can use a route reflector (which may be part of the controller). In the virtualized case, where a VPN forwarder is present, creation and configuration of VRFs, interfaces and installation of routes can be performed using a single protocol like XMPP, NC/YANG or an equivalent programmatic interface.

Also in the virtualized case, routes in the ingress and egress VRFs can be calculated by the controller based on its internal knowledge of the required SFC topology and the connectivity of SF instances to routing systems. In this case the routes are directly installed and no route advertisement is necessary.

As discussed further in Section 3, egress VRFs can load balance across the multiple next hops advertised from the next set of ingress VRFs.

Routes to the destination network via the first set of SF instances are advertised to the gateway router for the source network, and the egress VRFs of the last SF instance set have routes via the destination network gateway router.

2.6.2 Modified-Route SFC Creation

In this method of SFC configuration, all the VRFs connected to SF instances are configured with same import and export RT, so they form a VPN-connected mesh between the SF instance interfaces. This is termed the "Service VPN". A route is configured or learnt in each VRF with destination being the IP address of the connected SF instance via an interface configured in the VRF. The interface may be a physical or logical interface. The routing system that hosts such a VRF advertises a VPN route for each locally connected SF instance, with a forwarding label that enables it to forward incoming traffic from other routing systems to the connected SF instance. The VPN routes may be advertised via an RR or the controller, which then sends these updates to all the other routing systems that have VRFs with the service VPN RT. At this point all the VRFs have a route to reach every SF instance. The same IP address is used for each SF instance in a set, enabling load-balancing among multiple SF instances in the set.

The controller sends a route-policy to each routing system in the VPN, that describes the logical topology of each service chain that it belongs to. The route-policy contains entries in the form of a tuple for each service chain:

{Service-topology-name, Service-topology-RT, Service-node-sequence} where Service-node-sequence is simply an ordered list of the service function instance IP addresses that are in the chain.

Every service function chain has a single unique service-topology-RT that is provisioned on all participating routing systems in the relevant VRFs.

The VRF in the routing system that connects to the destination network is configured to attach the Service-topology-RT to exported routes, and the VRF in the gateway router of the source network will import routes using Service-topology-RT. A controller may also be used to originate the Service-topology-RT attached routes.

Route-policies may be described in a variety of formats in addition to that described above. For instance, it would be possible to use YANG as a modeling language.

Using Figure 1 for reference, when the gateway R-B advertises a VPN

route to Network-B, it attaches the Service-topology-RT. BGP route updates are sent to all the routing systems in the service VPN. The routing systems perform a modified set of actions for next-hop resolution and route installation in the ingress VRFs compared to normal BGP VPN behavior in routing systems, but no changes are required in the operation of the BGP protocol itself. The modification of behavior in the routing systems allows the automatic and constrained flow of traffic through the service chain.

Each routing system in the service VPN will process the VPN route to Network-B via R-B as follows:

1. If the routing system contains VRFs that import the Service-topology-RT, continue, otherwise ignore the route.

2. The routing system identifies the position and role (ingress/egress) of each of its VRFs in the SFC by comparing the IP address of the route in the VRF to the connected SF instance with those in the Service-node-sequence in the route-policy. Alternatively, the controller may provision the specific service node IP to be used as the next-hop in each VRF, in the route-policy.

3. The routing system modifies the next-hop of the imported route with the Service-topology-RT, to select the appropriate next-hop as per the route-policy. It ignores the next-hop and label in the received route. It resolves the selected next-hop in the VRF routing table.

   a. The imported route to Network-B in the ingress VRF is modified to have a next-hop of the IP address of the logically connected SF instance.

   b. The imported route to Network-B in the egress VRF is modified to have a next hop of the IP address of the next SF instance in the SFC.

4. The egress VRFs for the last service function install the VPN route via the gateway R-B unmodified.

Note that the modified routes are not re-advertised into the VPN by the various routing systems in the SFC.

Similar to the sequential VPN method, VRF configuration and creation, and routing-policy installation can be performed manually or via scripting, or a controller could be used to automate the process.

2.7 Controller Function

The purpose of the controller is to manage instantiation of SFCs in networks and datacenters. When an SFC is to be instantiated, a model of the desired topology (service functions, number of instances, connectivity) is built in the controller either via an API or GUI. The controller then selects resources in the infrastructure that will support the SFC and configures them. This can involve instantiation of SF instances to implement each service function, the instantiation of VRFs that will form virtual networks between SF instances, and installation of routes to cause traffic to flow into and between SF instances.

For simplicity, in this document, the controller is assumed to contain all the required features for management of SFCs. In actual implementations, these features may be distributed among multiple inter-connected systems. E.g. An overarching orchestrator might manage the overall SFC model, sending instructions to a separate virtual machine manager to instantiate service function instances, and to a virtual network manager to set up the service chain connections between them.

The controller can also perform necessary BGP signaling and route distribution actions as described throughout this document.

2.8 Variations on Setting Prefixes in an SFC

2.8.1 Variation 1

In the configuration methods described above, the network prefixes for each network (Network-A and Network-B in the example above) connected to the SFC are used in the routes that direct traffic through the SFC. This creates an operational linkage between the implementation of the SFC and the insertion of the SFC into a network.

For instance, subscriber network prefixes will normally be segmented across subscriber attachment points such as broadband or mobile gateways. This means that each SFC would have to be configured with the subscriber network prefixes whose traffic it is handling.

In a variation of the SFC configuration method described above, the prefixes used in each direction can be such that they include all possible addresses at each side of the SFC. For example, in Figure 1, the prefix for Network-A could include all subscriber IP addresses and the prefix for Network-B could be the default route, 0/0.

Using this technique, the same routes can be installed in all instances of an SFC that serve different groups of subscribers in different geographic locations.

The routes forwarding traffic into a SF instance and to the next SF instance are installed when an SFC is initially built, and each time a SF instance is connected into the SFC, but there is no requirement for VRFs to be reconfigured when traffic from different networks pass through the service chain, so long as their prefix is included in the prefixes in the VRFs along the SFC.

In this variation, it is assumed that no subscriber-originated traffic will enter the SFC destined for an IP address also in the subscriber network address range. This will not be a restriction in many cases.

2.8.2 Variation 2

As another slight variation of the above, a network prefix may be disaggregated and spread out among various gateway routers, for instance, in the case of virtual machines in a data-center. In order to reduce the scaling requirements on the routing systems along the SFC, the aggregate network prefix may be advertised with the Service-topology-RT and used in the traffic forwarding along the SFC.

Where there is a gateway router for the destination network that can aggregate the prefixes, none of the routing systems along the SFC need to receive the more-specific routes. If there is not, the service chain can be divided into two parts such that only the egress VRFs of the last SF instance import the more specific routes; and the rest of the VRFs only import the aggregate prefix. For instance, this may be done by using two different Service-topology-RTs for more-specific and aggregate routes.

In the simplest case, a default route is used to direct forwarding along the SFC upto the last SF instance, while the source network’s gateway routers and the egress VRF of the last SF instance use the destination network’s prefixes.

2.9 Header Transforming Service Functions

If a service function performs an action that changes the source address in the packet header (e.g., NAT), the routes that were installed as described above may not support reverse flow traffic. The solution to this is for the controller modify the routes in the reverse direction to direct traffic into instances of the transforming service function. The original routes with a source prefix (Network-A in Figure 2) are replaced with a route that has a

prefix that includes all the possible addresses that the source address could be mapped to. In the case of network address translation, this would correspond to the NAT pool.

3 Load Balancing Along a Service Function Chain

One of the key concepts driving NFV [NFVE2E]is the idea that each service function along an SFC can be separately scaled by changing the number of service function instances that implement it. This requires that load balancing be performed before entry into each service function. In this architecture, load balancing is performed in either or both of egress and ingress VRFs depending on the type of load balancing being performed, and if more than one service instance is connected to the same ingress VRF.

3.1 SF Instances Connected to Separate VRFs

If SF instances implementing a service in an SFC are each connected to separate VRFs(e.g. instances are connected to different routers or are running on different hosts), load balancing is performed in the egress VRFs of the previous service, or in the VRF that is the entry to the SFC. The controller distributes BGP multi-path routes to the egress VRFs. The destination prefix of each route is the ultimate destination network, or its representative aggregate or default. The next-hops in the ECMP set are BGP next-hops of the service instances attached to ingress VRFs of the next service in the SFC. The load balancing corresponds to BGP Multipath, which requires that the route distinguishers for each route are distinct in order to recognize that distinct paths should be used. Hence, each VRF in a distributed, SFC environment should have a unique route distinguisher.

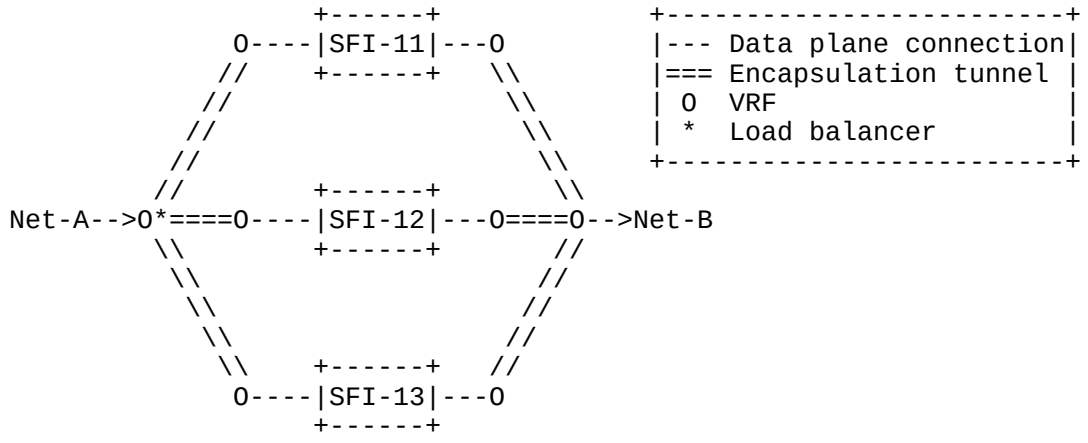


Figure 6 - Load Balancing across SF Instances Connected to Different VRFs

In the diagram, above, a service function is implemented in three service instances each connected to separate VRFs. Traffic from Network-A arrives at VRF at the start of the SFC, and is load balanced across the service instances using a set of ECMP routes with next hops being the addresses of the routing systems containing the ingress VRFs and with labels that identify the ingress interfaces of the service instances.

3.2 SF Instances Connected to the Same VRF

When SF instances implementing a service in an SFC are connected to the same ingress VRF, load balancing is performed in the ingress VRF across the service instances connected to it. The controller will install routes in the ingress VRF to the destination network with the interfaces connected to each service instance as next hops. The ingress VRF will then use ECMP to load balance across the service instances.

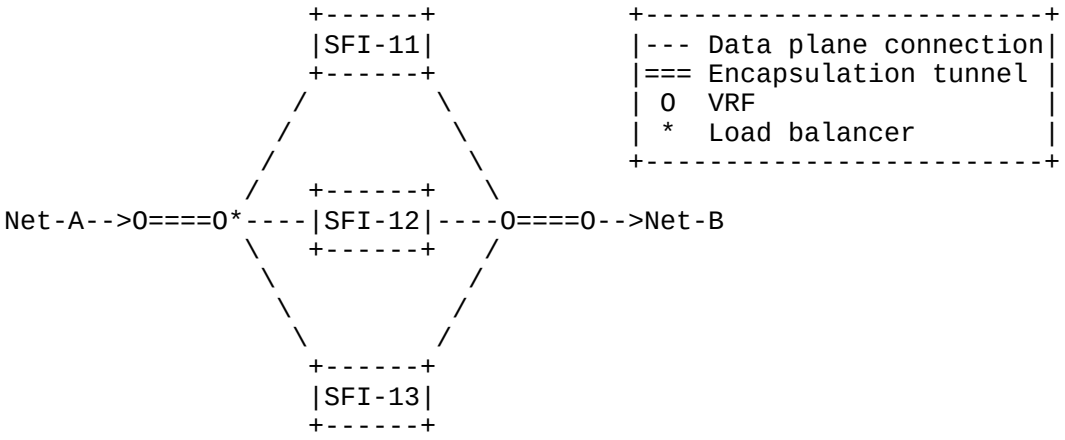


Figure 7 - Load Balancing across SF Instances Connected to the Same VRF

In the diagram, above, a service is implemented by three service instances that are connected to the same ingress and egress VRFs. The ingress VRF load balances across the ingress interfaces using ECMP, and the egress traffic is aggregated in the egress VRF.

3.3 Combination of Egress and Ingress VRF Load Balancing

In Figure 8, below, an example SFC is shown where load balancing is performed in both ingress and egress VRFs.

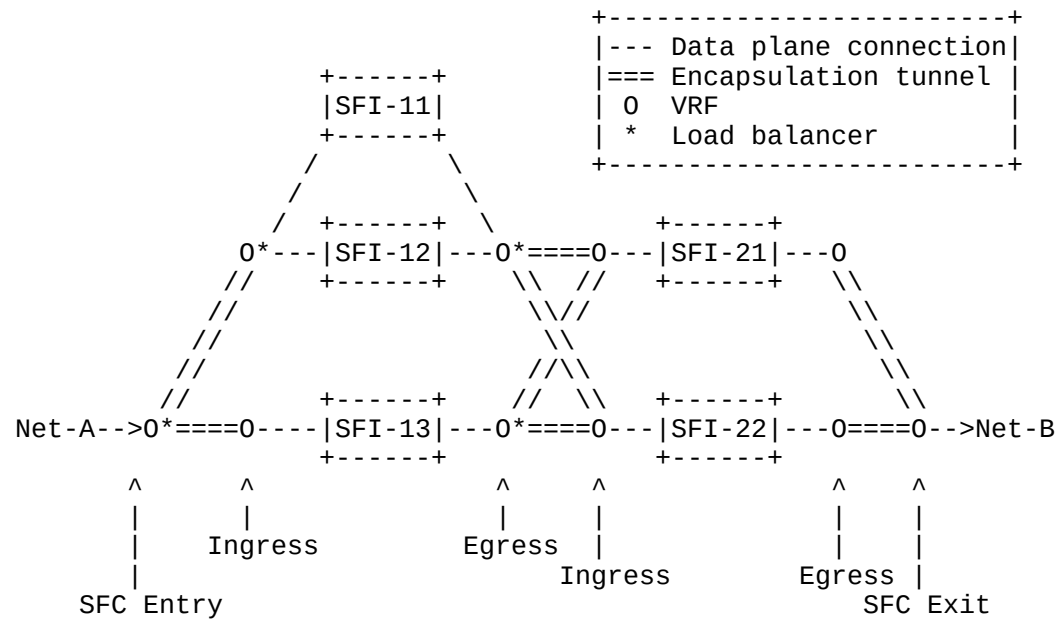


Figure 8 - Load Balancing across SF Instances

In Figure 8, above, an SFC is composed of two services implemented by three service instances and two service instances, respectively. The service instances SFI-11 and SFI-12 are connected to the same ingress and egress VRFs, and all the other service instances are connected to separate VRFs.

Traffic entering the SFC from Network-A is load balanced across the ingress VRFs of the first service function by the chain entry VRF, and then load balanced again across the ingress interfaces of SFI-11 and SFI-12 by the shared ingress VRF. Note that use of standard ECMP will lead to an uneven distribution of traffic between the three service instances (25% to SFI-11, 25% to SFI-12, and 50% to SFI-13). This issue can be mitigated through the use of BGP link bandwidth extended community [draft-ietf-idr-link-bandwidth].

After traffic passes through the first set of service instances, it is load balanced in each of the egress VRFs of the first set of service instances across the ingress VRFs of the next set of service

instances.

3.4 Forward and Reverse Flow Load Balancing

This section discusses requirements in load balancing for forward and reverse paths when stateful service functions are deployed.

3.4.1 Issues with Equal Cost Multi-Path Routing

As discussed in the previous sections, load balancing in the forward SFC in the above example can automatically occur with standard BGP, if multiple equal cost routes to Network-B are installed into all the ingress VRFs, and each route directs traffic through a different service function instance in the next set. The multiple BGP routes in the routing table will translate to Equal Cost Multi-Path in the forwarding table. The hash used in the load balancing algorithm (per packet, per flow or per prefix) is implementation specific.

If a service function is stateful, it is required that forward flows and reverse flows always pass through the same service function instance. ECMP does not provide this capability, since the hash calculation will see different input data for the same flow in the forward and reverse directions (since the source and destination fields are reversed).

Additionally, if the number of SF instances changes, either increasing to expand capacity, or decreases (planned, or due to a SF instance failure), the hash table in ECMP is recalculated, and most flows will be directed to a different SF instance and user sessions will be disrupted.

There are a number of ways to satisfy the requirements of symmetric forward/reverse paths for flows and minimal disruption when SF instances are added to or removed from a set. Two techniques that can be employed are described in the following sections.

3.4.2 Modified ECMP with Consistent Hash

Symmetric forwarding into each side of an SF instance set can be achieved with a small modification to ECMP if the packet headers are preserved after passing through a SF instance set. In this case, each packet’s 5-tuple data can be used in a hashing function, provided the source and destination IP address and port information are swapped in the reverse calculation and that the same or no hash salt is used for both directions. This method only requires that the list of available service function instances is consistently maintained in all the load balancers, rather than maintaining a distributed flow table.

In the SFC architecture described in this document, when SF instances are added or removed, the controller is required to configure (or remove) static routes to the SF instances. The controller could configure the load balancing function in VRFs that connect to each added (or removed) SF instance as part of the same network transaction as route updates to ensure that the load balancer configuration is synchronized with the set of SF instances.

The effect of rehashing when SF instances are added or removed can be minimized, or even eliminated using variations of the technique of consistent hashing [consistent-hash]. Details are outside the scope of this document.

3.4.3 ECMP with Flow Table

A second refinement that can ensure forward/reverse flow consistency, and also provides stability when the number of SF instances changes ("flow-stickiness"), is the use of dynamically configured IP flow tables in the VRFs. In this technique, flow tables are used to ensure that existing flows are unaffected if the number of ECMP routes changes, and that forward and reverse traffic passes through the same SF instance in each set of SF instances implementing a service function.

The flow tables are set up as follows:

1. User traffic with a new 5-tuple enters an egress VRF from a connected SF instance.

2. The VRF calculates the ECMP hash across available routes (i.e., ECMP group) to the ingress interfaces of the SF instances in the next SF instance set.

3. The VRF creates a new flow entry for the 5-tuple traffic with the next-hop being the chosen downstream ECMP group member (determined in the step 2. above) . All subsequent packets for the same flow will be forwarded using flow lookup and, hence, will use the same next-hop.

4. The encapsulated packet arrives in the routing system that hosts the ingress VRF for the selected SF instance.

5. The ingress VRF of the next service instance determines if the packet came from a routing system that is in an ECMP group in the reverse direction(i.e., from this ingress VRF back to the previous set of SF instances).

6. If an ECMP group is found, the ingress VRF creates a reverse flow entry for the 5-tuple with next-hop of the tunnel on which

traffic arrived.

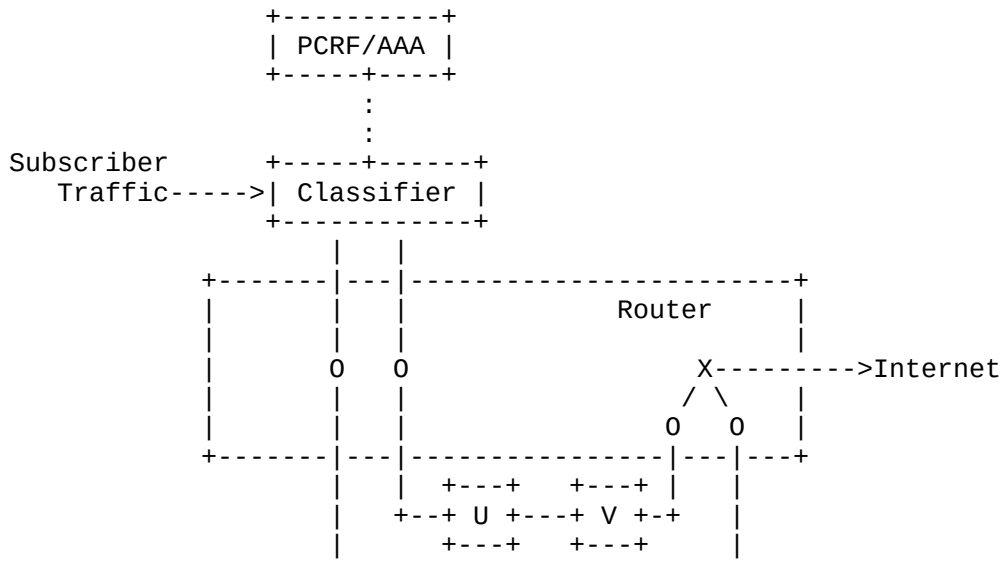
7. The packet is sent into the SF instance connected to the ingress VRF.

The above method ensures that forward and reverse flows pass through the same SF instances, and that if the number of ECMP routes changes when SF instances are added or removed, all existing flows will continue to flow through the same SF instances, but new flows will use the new ECMP hash. The only flows affected will be those that were passing through an SF instance that was removed, and those will be spread among the remaining SF instances using the updated ECMP hash.

4 Steering into SFCs Using a Classifier

In many applications of SFCs, a classifier will be used to direct traffic into SFCs. The classifier inspects the first or first few packets in a flow to determine which SFC the flow should be sent into. The decision criteria can include the IP 5-tuple of the header, and/or analysis of the payload of packets using deep packet inspection. Integration with a subscriber management system such as PCRF or AAA will usually be required in order to identify which SFC to send traffic to based on subscriber policy.

An example logical architecture is shown in Figure 9, below where a classifier is external to a physical router.



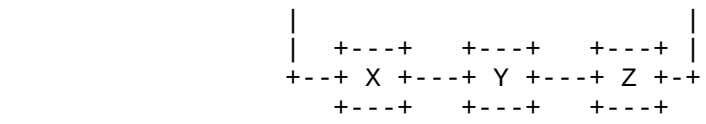


Figure 9 - Subscriber/Application-Aware Steering with a Classifier

In the diagram, the classifier receives subscriber traffic and sends the traffic out of one of two logical interfaces, depending on classification criteria. The logical interfaces of the classifier are connected to VRFs in a router that are entries to two SFCs (shown as O in the diagram).

In this scenario, the exit VRF for each SFC does not peer with a gateway or proxy node in the destination network and packets are forwarded using IP lookup in the main routing table or in a VRF that the exit traffic from the SFCs is directed into (shown as X in the diagram).

An alternative would be where the classifier is itself a distributed, virtualized service function, but with multiple egress interfaces. In that case, each virtual classifier instance could be attached to a set of VRFs that connect to different SFCs. Each chain entry VRF would load balance across the first SF instance set in its SFC. The reverse flow table mechanism described in Section 3.4.3 could be employed to ensure that flows return to the originating classifier instance which may maintain subscriber context and perform charging and accounting.

5 External Domain Co-ordination

It is likely that SFCs will be managed as a separate administrative domain from the networks that they receive traffic from, and send traffic to. If the connected networks use BGP for route distribution, the controller in the SFC domain can join the network domains by creating BGP peering sessions with routing systems or route reflectors in those network domains.

In order to steer traffic from the network domains into an SFC, the controller will advertise a destination network’s prefixes into the peering network domain with a BGP next-hop and label associated with the SFC entry point, that may be on a routing system attached to the first SF instance. This advertisement may be over regular MP-BGP/VPN peering which assumes existing standard VPN routing/forwarding behavior on the network domain’s routers (PEs/ASBRs).

An operational benefit of this approach is also that the SFC topology within a domain need not be exposed to other domains.

6 Fine-grained steering using BGP Flow-Spec

When steering traffic from a network domain's existing routing systems into an SFC is desired based on attributes of the packet flow, [FLOWSPEC] is a signaling option that can be used. In this case, the controller advertises a flow-spec route to the network domain's routing systems or route reflectors with the appropriate next-hop or Service-topology-RT for the SFC entry point.

7 BGP-EVPN signaling

In a DC environment, routing systems are likely to use VXLAN based overlays and a BGP EVPN control plane (DC-OVERLAY). For the solution designs described earlier in the document, the BGP VPN routes for both the SF instances and the destination networks are advertised via BGP-EVPN, using type-2 and type-5 route types.

8 Controller Federation

When SFCs are distributed geographically, or in very large-scale environments, there may be multiple SFC controllers present. If there is a requirement for SFCs to span controller domains there may be a requirement to exchange information between controllers. Again, a BGP session between controllers can be used to exchange route information as described in the previous sections and allow such domain spanning SFCs to be created.

9 Summary and Conclusion

The architecture for service function chains described in this document uses virtual networks implemented as overlays in order to create service function chains. The virtual networks use standards-based encapsulation tunneling, such as MPLS over GRE/UDP or VXLAN, to transport packets into an SFC and between service function instances without routing in the user address space. Two methods of installing routes to form service chains are described.

In environments with physical routers, a controller may operate in tandem with existing BGP route reflectors, and would contain the SFC topology model, and the ability to install the local static interface routes to SF instances. In a virtualized environment, the controller can emulate route refection internally and simply install required routes directly without advertisements occurring.

10 Security Considerations

The security considerations for SFCs are broadly similar to those concerning the data, control and management planes of any device placed in a network. Details are out of scope for this document.

11 IANA Considerations

There are no IANA considerations.

12 Acknowledgements

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

This document is a merged specification based on earlier drafts [draft-rfernando-bess-service-chaining] and [draft-mackie-sfc-using-virtual-networking].

The authors would like to thank D. Daino, D.R. Lopez, D. Bernier, W. Haeffner, A. Farrel, L. Fang, and N. So, for their contributions to the earlier drafts. The authors would also like to thank the following individuals for their review and feedback on the original proposals: E. Rosen, J. Guchard, P. Quinn, P. Bosch, D. Ward, A. Ganesan, T. Morin, N. Seth, G. Pildush and N. Bitar.

13 References

13.1 Normative References

None

13.2 Informative References

[NFVE2E] "Network Functions Virtualisation: End to End Architecture, http://docbox.etsi.org/ISG/NFV/70-DRAFT/0010/NFV-0010v016.zip".

[RFC2328] J. Moy, "OSPF Version 2", RFC 2328, April, 1998.

[draft-merged-sfc-architecture] Halpern, J. and Pignataro, C., "Service Function Chaining (SFC) Architecture", draft-ietf-sfc-architecture-09 June 2015.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

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

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

[RFC7348] Mahalingam, M., et al. "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks.", RFC 7348, August 2014.

[draft-ietf-l3vpn-end-system] Marques, P., et al., "BGP-signaled end-system IP/VPNs", draft-ietf-l3vpn-end-system-04, October 2, 2014.

[FLOWSPEC] Marques, P., Sheth, N., Raszuk, R., et al., "Dissemination of Flow Specification Rules", RFC 5575, August 2009.

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

[draft-ietf-sfc-nsh] Quinn, P., et al, "Network Service Header", draft-ietf-sfc-nsh-00, March 2015.

[draft-niu-sfc-mechanism] Niu, L., Li, H., and Jiang, Y., "A Service Function Chaining Header and its Mechanism", draft-niu-sfc-mechanism-00, January 2014.

[draft-rijsman-sfc-metadata-considerations] B. Rijsman, et al. "Metadata Considerations", draft-rijsman-sfc-metadata-considerations-00, February 12, 2014

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC4023] Worster, T., Rekhter, Y., and E. Rosen, "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, March 2005.

[RFC7510] Xu, X., Sheth, N. et al, "Encapsulating MPLS in UDP", RFC 7510, April 2015.

[draft-ietf-i2rs-architecture] Atlas, A., Halpern, J., Hares, S., Ward, D., and T Nadeau, "An Architecture for the Interface to the Routing System", draft-ietf-i2rs-architecture, work in progress, March 2015.

[consistent-hash] Karger, D.; Lehman, E.; Leighton, T.; Panigrahy, R.; Levine, M.; Lewin, D. (1997). "Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web". Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing. ACM Press New York, NY, USA. pp. 654-663.

[draft-ietf-idr-link-bandwidth] P. Mohapatra, R. Fernando, "BGP Link Bandwidth Extended Community", draft-ietf-idr-link-

bandwidth, work in progress.

[I-D.fang-l3vpn-virtual-pe]
L. Fang, et al., "BGP/MPLS IP VPN Virtual PE",
draft-fang-l3vpn-virtual-pe, work in progress.

[I-D.ietf-i2rs-problem-statement]
Atlas, A., Nadeau, T., and D. Ward, "Interface to the Routing System Problem Statement",
draft-ietf-i2rs-problem-statement, work in progress.

Authors' Addresses

Rex Fernando
Cisco
170 W Tasman Dr
San Jose, CA
Email: rex@cisco.com

Stuart Mackie
Juniper Networks
1133 Innovation Way
Sunnyvale, CA 94089
USA
Email: wsmackie@juniper.net

Dhananjaya Rao
Cisco
170 W Tasman Dr
San Jose, CA
Email: dhrao@cisco.com

Bruno Rijsman
Juniper Networks
1133 Innovation Way
Sunnyvale, CA 94089
USA
Email: brijsman@juniper.net

Maria Napierala
AT&T
200 Laurel Avenue
Middletown, NJ 07748
Email: mnapierala@att.com

BESS Workgroup
Internet Draft
Intended status: Standards Track

J. Rabadan
S. Sathappan
W. Henderickx
S. Palislamovic
Alcatel-Lucent

R. Shekhar
A. Lohiya
Juniper

A. Sajassi
D. Cai
Cisco

Expires: January 7, 2016

July 6, 2015

Interconnect Solution for EVPN Overlay networks
draft-ietf-bess-dci-evpn-overlay-01

Abstract

This document describes how Network Virtualization Overlay networks (NVO) can be connected to a Wide Area Network (WAN) in order to extend the layer-2 connectivity required for some tenants. The solution analyzes the interaction between NVO networks running EVPN and other L2VPN technologies used in the WAN, such as VPLS/PBB-VPLS or EVPN/PBB-EVPN, and proposes a solution for the interworking between both.

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.

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

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

The list of Internet-Draft Shadow Directories can be accessed at

http://www.ietf.org/shadow.html

This Internet-Draft will expire on January 7, 2016.

Table of Contents

1. Introduction

[EVPN-Overlays] discusses the use of EVPN as the control plane for Network Virtualization Overlay (NVO) networks, where VXLAN, NVGRE or MPLS over GRE can be used as possible data plane encapsulation options.

While this model provides a scalable and efficient multi-tenant solution within the Data Center, it might not be easily extended to the WAN in some cases due to the requirements and existing deployed technologies. For instance, a Service Provider might have an already deployed (PBB-)VPLS or (PBB-)EVPN network that must be used to interconnect Data Centers and WAN VPN users. A Gateway (GW) function is required in these cases.

This document describes a Interconnect solution for EVPN overlay networks, assuming that the NVO Gateway (GW) and the WAN Edge functions can be decoupled in two separate systems or integrated into the same system. The former option will be referred as "Decoupled Interconnect solution" throughout the document whereas the latter one will be referred as "Integrated Interconnect solution".

2. Decoupled Interconnect solution for EVPN overlay networks

This section describes the interconnect solution when the GW and WAN Edge functions are implemented in different systems. Figure 1 depicts the reference model described in this section.

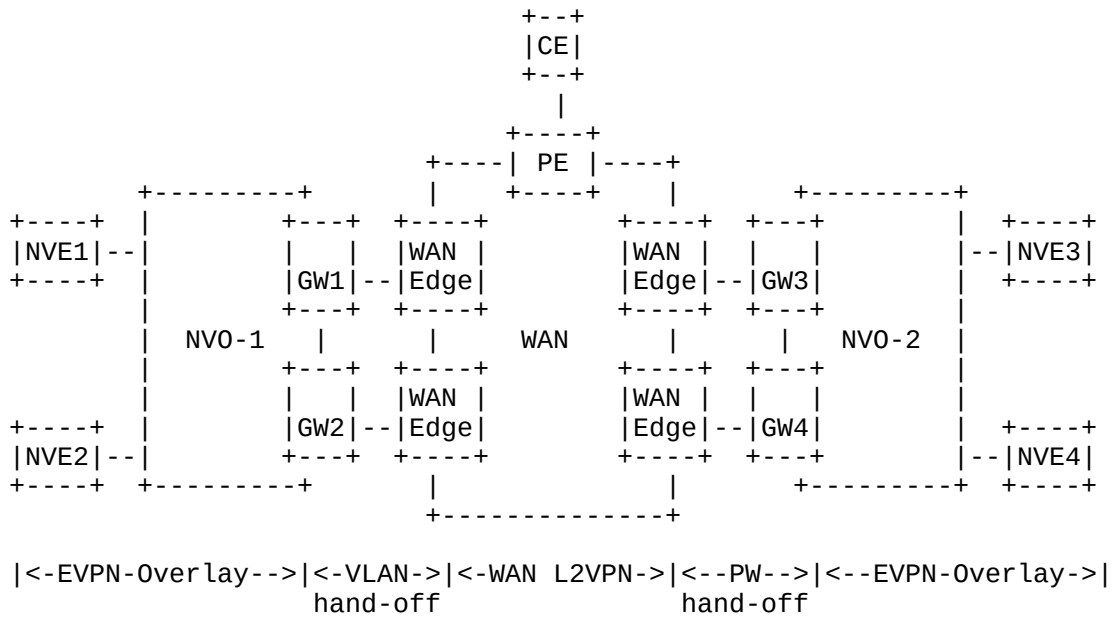


Figure 1 Decoupled Interconnect model

The following section describes the interconnect requirements for this model.

2.1. Interconnect requirements

This proposed Interconnect architecture will be normally deployed in networks where the EVPN-Overlay and WAN providers are different entities and a clear demarcation is needed. The solution must observe the following requirements:

- o A simple connectivity hand-off must be provided between the EVPN-Overlay network provider and the WAN provider so that QoS and security enforcement are easily accomplished.

- o The solution must be independent of the L2VPN technology deployed in the WAN.

- o Multi-homing between GW and WAN Edge routers is required. Per-service load balancing MUST be supported. Per-flow load balancing MAY be supported but it is not a strong requirement since a deterministic path per service is usually required for an easy QoS and security enforcement.

- o Ethernet OAM and Connectivity Fault Management (CFM) functions must

be supported between the EVPN-Overlay network and the WAN network.

- o The following optimizations MAY be supported at the GW:
  - + Flooding reduction of unknown unicast traffic sourced from the DC Network Virtualization Edge devices (NVEs).
  - + Control of the WAN MAC addresses advertised to the DC.
  - + ARP flooding control for the requests coming from the WAN.

2.2. VLAN-based hand-off

In this option, the hand-off between the GWs and the WAN Edge routers is based on 802.1Q VLANs. This is illustrated in Figure 1 (between the GWs in NVO-1 and the WAN Edge routers). Each MAC-VRF in the GW is connected to a different VSI/MAC-VRF instance in the WAN Edge router by using a different C-TAG VLAN ID or a different combination of S/C-TAG VLAN IDs that matches at both sides.

This option provides the best possible demarcation between the DC and WAN providers and it does not require control plane interaction between both providers. The disadvantage of this model is the provisioning overhead since the service must be mapped to a S/C-TAG VLAN ID combination at both, GW and WAN Edge routers.

In this model, the GW acts as a regular Network Virtualization Edge (NVE) towards the DC. Its control plane, data plane procedures and interactions are described in [EVPN-Overlays].

The WAN Edge router acts as a (PBB-)VPLS or (PBB-)EVPN PE with attachment circuits (ACs) to the GWs. Its functions are described in [RFC4761][RFC4762][RFC6074] or [RFC7432][PBB-EVPN].

2.3. PW-based (Pseudowire-based) hand-off

If MPLS can be enabled between the GW and the WAN Edge router, a PW-based Interconnect solution can be deployed. In this option the hand-off between both routers is based on FEC128-based PWs or FEC129-based PWs (for a greater level of network automation). Note that this model still provides a clear demarcation boundary between DC and WAN, and security/QoS policies may be applied on a per PW basis. This model provides better scalability than a C-TAG based hand-off and less provisioning overhead than a combined C/S-TAG hand-off. The PW-based hand-off interconnect is illustrated in Figure 1 (between the NVO-2 GWs and the WAN Edge routers).

In this model, besides the usual MPLS procedures between GW and WAN Edge router, the GW MUST support an interworking function in each MAC-VRF that requires extension to the WAN:

- o If a FEC128-based PW is used between the MAC-VRF (GW) and the VSI (WAN Edge), the provisioning of the VCID for such PW MUST be supported on the MAC-VRF and must match the VCID used in the peer VSI at the WAN Edge router.

- o If BGP Auto-discovery [RFC6074] and FEC129-based PWs are used between the GW MAC-VRF and the WAN Edge VSI, the provisioning of the VPLS-ID MUST be supported on the MAC-VRF and must match the VPLS-ID used in the WAN Edge VSI.

2.4. Multi-homing solution on the GWs

As already discussed, single-active multi-homing, i.e. per-service load-balancing multi-homing MUST be supported in this type of interconnect. All-active multi-homing may be considered in future revisions of this document.

The GWs will be provisioned with a unique ESI per WAN interconnect and the hand-off attachment circuits or PWs between the GW and the WAN Edge router will be assigned to such ESI. The ESI will be administratively configured on the GWs according to the procedures in [RFC7432]. This Interconnect ESI will be referred as "I-ESI" hereafter.

The solution (on the GWs) MUST follow the single-active multi-homing procedures as described in [EVPN-Overlays] for the provisioned I-ESI, i.e. Ethernet A-D routes per ESI and per EVI will be advertised to the DC NVEs. The MAC addresses learnt (in the data plane) on the hand-off links will be advertised with the I-ESI encoded in the ESI field.

2.5. Gateway Optimizations

The following features MAY be supported on the GW in order to optimize the control plane and data plane in the DC.

2.5.1 Use of the Unknown MAC route to reduce unknown flooding

The use of EVPN in the NVO networks brings a significant number of benefits as described in [EVPN-Overlays]. There are however some potential issues that SHOULD be addressed when the DC EVIs are connected to the WAN VPN instances.

The first issue is the additional unknown unicast flooding created in the DC due to the unknown MACs existing beyond the GW. In virtualized DCs where all the MAC addresses are learnt in the control/management plane, unknown unicast flooding is significantly reduced. This is no longer true if the GW is connected to a layer-2 domain with data

plane learning.

The solution suggested in this document is based on the use of an "Unknown MAC route" that is advertised by the Designated Forwarder GW. The Unknown MAC route is a regular EVPN MAC/IP Advertisement route where the MAC Address Length is set to 48 and the MAC address to 00:00:00:00:00:00 (IP length is set to 0).

If this procedure is used, when an EVI is created in the GWs and the Designated Forwarder (DF) is elected, the DF will send the Unknown MAC route. The NVEs supporting this concept will prune their unknown unicast flooding list and will only send the unknown unicast packets to the owner of the Unknown MAC route. Note that the I-ESI will be encoded in the ESI field of the NLRI so that regular multi-homing procedures can be applied to this unknown MAC too (e.g. backup-path).

2.5.2. MAC address advertisement control

Another issue derived from the EVI interconnect to the WAN layer-2 domain is the potential massive MAC advertisement into the DC. All the MAC addresses learnt from the WAN on the hand-off attachment circuits or PWs must be advertised by BGP EVPN. Even if optimized BGP techniques like RT-constraint are used, the amount of MAC addresses to advertise or withdraw (in case of failure) from the GWs can be difficult to control and overwhelming for the DC network, especially when the NVEs reside in the hypervisors.

This document proposes the addition of administrative options so that the user can enable/disable the advertisement of MAC addresses learnt from the WAN as well as the advertisement of the Unknown MAC route from the DF GW. In cases where all the DC MAC addresses are learnt in the control/management plane, the GW may disable the advertisement of WAN MAC addresses. Any frame with unknown destination MAC will be exclusively sent to the Unknown MAC route owner(s).

2.5.3. ARP flooding control

Another optimization mechanism, naturally provided by EVPN in the GWs, is the Proxy ARP/ND function. The GWs SHOULD build a Proxy ARP/ND cache table as per [RFC7432]. When the active GW receives an ARP/ND request/solicitation coming from the WAN, the GW does a Proxy ARP/ND table lookup and replies as long as the information is available in its table.

This mechanism is especially recommended on the GWs since it protects the DC network from external ARP/ND-flooding storms.

2.5.4. Handling failures between GW and WAN Edge routers

Link/PE failures MUST be handled on the GWs as specified in [RFC7432]. The GW detecting the failure will withdraw the EVPN routes as per [RFC7432].

Individual AC/PW failures should be detected by OAM mechanisms. For instance:

- If the Interconnect solution is based on a VLAN hand-off, 802.1ag/Y.1731 Ethernet-CFM MAY be used to detect individual AC failures on both, the GW and WAN Edge router. An individual AC failure will trigger the withdrawal of the corresponding A-D per EVI route as well as the MACs learnt on that AC.

- If the Interconnect solution is based on a PW hand-off, the LDP PW Status bits TLV MAY be used to detect individual PW failures on both, the GW and WAN Edge router.

3. Integrated Interconnect solution for EVPN overlay networks

When the DC and the WAN are operated by the same administrative entity, the Service Provider can decide to integrate the GW and WAN Edge PE functions in the same router for obvious CAPEX and OPEX saving reasons. This is illustrated in Figure 2. Note that this model does not provide an explicit demarcation link between DC and WAN anymore.

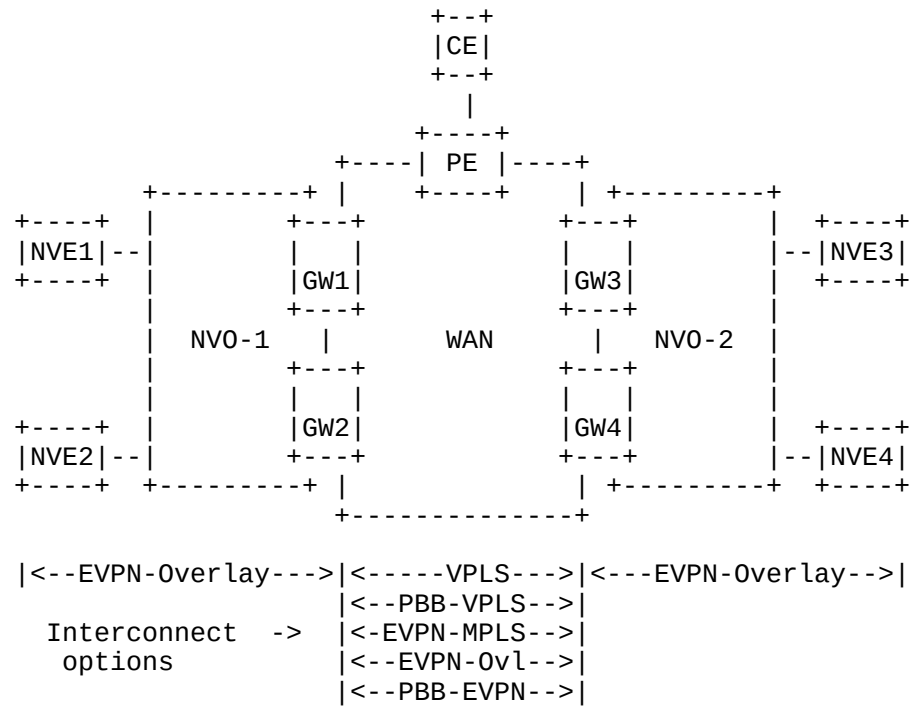


Figure 2 Integrated Interconnect model

3.1. Interconnect requirements

The solution must observe the following requirements:

- The GW function must provide control plane and data plane interworking between the EVPN-overlay network and the L2VPN technology supported in the WAN, i.e. (PBB-)VPLS or (PBB-)EVPN, as depicted in Figure 2.

- Multi-homing MUST be supported. Single-active multi-homing with per-service load balancing MUST be implemented. All-active multi-homing, i.e. per-flow load-balancing, MUST be implemented as long as the technology deployed in the WAN supports it.

- If EVPN is deployed in the WAN, the MAC Mobility, Static MAC protection and other procedures (e.g. proxy-arp) described in [RFC7432] must be supported end-to-end.

- Any type of inclusive multicast tree MUST be independently supported in the WAN as per [RFC7432], and in the DC as per [EVPN-Overlays].

3.2. VPLS Interconnect for EVPN-Overlay networks

3.2.1. Control/Data Plane setup procedures on the GWs

Regular MPLS tunnels and TLDP/BGP sessions will be setup to the WAN PEs and RRs as per [RFC4761][RFC4762][RFC6074] and overlay tunnels and EVPN will be setup as per [EVPN-Overlays]. Note that different route-targets for the DC and for the WAN are normally required. A single type-1 RD per service can be used.

In order to support multi-homing, the GWs will be provisioned with an I-ESI (see section 2.4), that will be unique per interconnection. All the [RFC7432] procedures are still followed for the I-ESI, e.g. any MAC address learnt from the WAN will be advertised to the DC with the I-ESI in the ESI field.

A MAC-VRF per EVI will be created in each GW. The MAC-VRF will have two different types of tunnel bindings instantiated in two different split-horizon-groups:

- VPLS PWs will be instantiated in the "WAN split-horizon-group".
- Overlay tunnel bindings (e.g. VXLAN, NVGRE) will be instantiated in the "DC split-horizon-group".

Attachment circuits are also supported on the same MAC-VRF, but they will not be part of any of the above split-horizon-groups.

Traffic received in a given split-horizon-group will never be forwarded to a member of the same split-horizon-group.

As far as BUM flooding is concerned, a flooding list will be created with the sub-list created by the inclusive multicast routes and the sub-list created for VPLS in the WAN. BUM frames received from a local attachment circuit will be flooded to both sub-lists. BUM frames received from the DC or the WAN will be forwarded to the flooding list observing the split-horizon-group rule described above.

Note that the GWs are not allowed to have an EVPN binding and a PW to the same far-end within the same MAC-VRF in order to avoid loops and packet duplication. This is described in [EVPN-VPLS-INTEGRATION].

The optimizations procedures described in section 2.5 can also be applied to this model.

3.2.2. Multi-homing procedures on the GWs

Single-active multi-homing MUST be supported on the GWs. All-active multi-homing is not supported by VPLS.

All the single-active multi-homing procedures as described by [EVPN-Overlays] will be followed for the I-ESI.

The non-DF GW for the I-ESI will block the transmission and reception of all the bindings in the "WAN split-horizon-group" for BUM and unicast traffic.

3.3. PBB-VPLS Interconnect for EVPN-Overlay networks

3.3.1. Control/Data Plane setup procedures on the GWs

In this case, there is no impact on the procedures described in [RFC7041] for the B-component. However the I-component instances become EVI instances with EVPN-Overlay bindings and potentially local attachment circuits. M MAC-VRF instances can be multiplexed into the same B-component instance. This option provides significant savings in terms of PWs to be maintained in the WAN.

The I-ESI concept described in section 3.2.1 will also be used for the PBB-VPLS-based Interconnect.

B-component PWs and I-component EVPN-overlay bindings established to the same far-end will be compared. The following rules will be observed:

- Attempts to setup a PW between the two GWs within the B-component context will never be blocked.

- If a PW exists between two GWs for the B-component and an attempt is made to setup an EVPN binding on an I-component linked to that B-component, the EVPN binding will be kept operationally down. Note that the BGP EVPN routes will still be valid but not used.

- The EVPN binding will only be up and used as long as there is no PW to the same far-end in the corresponding B-component. The EVPN bindings in the I-components will be brought down before the PW in the B-component is brought up.

The optimizations procedures described in section 2.5 can also be applied to this Interconnect option.

3.3.2. Multi-homing procedures on the GWs

Single-active multi-homing MUST be supported on the GWs.

All the single-active multi-homing procedures as described by [EVPN-Overlays] will be followed for the I-ESI for each EVI instance connected to B-component.

3.4. EVPN-MPLS Interconnect for EVPN-Overlay networks

If EVPN for MPLS tunnels, EVPN-MPLS hereafter, is supported in the WAN, an end-to-end EVPN solution can be deployed. The following sections describe the proposed solution as well as the impact required on the [RFC7432] procedures.

3.4.1. Control Plane setup procedures on the GWs

The GWs MUST establish separate BGP sessions for sending/receiving EVPN routes to/from the DC and to/from the WAN. Normally each GW will setup one (two) BGP EVPN session(s) to the DC RR(s) and one(two) session(s) to the WAN RR(s). The same route-distinguisher (RD) per MAC-VRF can be used for the EVPN service routes sent to both, WAN and DC RRs. On the contrary, although reusing the same value is possible, different route-targets are expected to be handled for the same EVI in the WAN and the DC. Note that the EVPN service routes sent to the DC RRs will normally include a [RFC5512] BGP encapsulation extended community with a different tunnel type than the one sent to the WAN RRs.

As in the other discussed options, an I-ESI will be configured on the GWs for multi-homing. This I-ESI represents the WAN to the DC but also the DC to the WAN.

Received EVPN routes will never be reflected on the GWs but consumed and re-advertised (if needed):

- Ethernet A-D routes, ES routes and Inclusive Multicast routes are consumed by the GWs and processed locally for the corresponding [RFC7432] procedures.

- MAC/IP advertisement routes will be received, imported and if they become active in the MAC-VRF MAC FIB, the information will be re-advertised as new routes with the following fields:

  + The RD will be the GW’s RD for the MAC-VRF.

  + The ESI will be set to the I-ESI.

  + The Ethernet-tag value will be kept from the received NLRI.

  + The MAC length, MAC address, IP Length and IP address values will be kept from the received NLRI.

+ The MPLS label will be a local 20-bit value (when sent to the WAN) or a DC-global 24-bit value (when sent to the DC).

+ The appropriate Route-Targets (RTs) and [RFC5512] BGP Encapsulation extended community will be used according to [EVPN-Overlays].

The GWs will also generate the following local EVPN routes that will be sent to the DC and WAN, with their corresponding RTs and [RFC5512] BGP Encapsulation extended community values:

- o ES route for the I-ESI.
- o Ethernet A-D routes per ESI and EVI for the I-ESI. The A-D per-EVI routes sent to the WAN and the DC will have a consistent Ethernet-Tag values.
- o Inclusive Multicast routes with independent tunnel type value for the WAN and DC. E.g. a P2MP LSP may be used in the WAN whereas ingress replication may be used in the DC. The routes sent to the WAN and the DC will have a consistent Ethernet-Tag.
- o MAC/IP advertisement routes for MAC addresses learned in local attachment circuits. Note that these routes will not include the I-ESI, but ESI=0 or different from 0 for local Ethernet Segments (ES). The routes sent to the WAN and the DC will have a consistent Ethernet-Tag.

Assuming GW1 and GW2 are peer GWs of the same DC, each GW will generate two sets of local service routes: Set-DC will be sent to the DC RRs and will include A-D per EVI, Inclusive Multicast and MAC/IP routes for the DC encapsulation and RT. Set-WAN will be sent to the WAN RRs and will include the same routes but using the WAN RT and encapsulation. GW1 and GW2 will receive each other’s set-DC and set-WAN. This is the expected behavior on GW1 and GW2 for locally generated routes:

- o Inclusive multicast routes: when setting up the flooding lists for a given MAC-VRF, each GW will include its DC peer GW only in the EVPN-overlay flooding list (by default) and not the EVPN-MPLS flooding list. That is, GW2 will import two Inclusive Multicast routes from GW1 (from set-DC and set-WAN) but will only consider one of the two, having the set-DC route higher priority.
- o MAC/IP advertisement routes for local attachment circuits: as above, the GW will select only one, having the route from the set-DC a higher priority.

3.4.2. Data Plane setup procedures on the GWs

The procedure explained at the end of the previous section will make sure there are no loops or packet duplication between the GWs of the same DC (for frames generated from local ACs) since only one EVPN binding per EVI will be setup in the data plane between the two nodes. That binding will by default be added to the EVPN-overlay flooding list.

As for the rest of the EVPN tunnel bindings, they will be added to one of the two flooding lists that each GW sets up for the same MAC-VRF:

- EVPN-overlay flooding list (composed of bindings to the remote NVEs or multicast tunnel to the NVEs).
- EVPN-MPLS flooding list (composed of MP2P or LSM tunnel to the remote PEs)

Each flooding list will be part of a separate split-horizon-group: the WAN split-horizon-group or the DC split-horizon-group. Traffic generated from a local AC can be flooded to both split-horizon-groups. Traffic from a binding of a split-horizon-group can be flooded to the other split-horizon-group and local ACs, but never to a member of its own split-horizon-group.

When either GW1 or GW2 receive a BUM frame on an overlay tunnel, they will perform a tunnel IP SA lookup to determine if the packet’s origin is the peer DC GW, i.e. GW2 or GW1 respectively. If the packet is coming from the peer DC GW, it MUST only be flooded to local attachment circuits and not to the WAN split-horizon-group (the assumption is that the peer GW would have sent the BUM packet to the WAN directly).

3.4.3. Multi-homing procedures on the GWs

Single-active as well as all-active multi-homing MUST be supported.

All the multi-homing procedures as described by [RFC7432] will be followed for the DF election for I-ESI, as well as the backup-path (single-active) and aliasing (all-active) procedures on the remote PEs/NVEs. The following changes are required at the GW with respect to the I-ESI:

- Single-active multi-homing; assuming a WAN split-horizon-group, a DC split-horizon-group and local ACs on the GWs:

+ Forwarding behavior on the non-DF: the non-DF MUST NOT forward BUM or unicast traffic received from a given split-horizon-group to a member of its own split-horizon-group or to the other split-horizon-group. Only forwarding to local ACs is allowed (as long as they are not part of an ES for which the node is non-DF).

+ Forwarding behavior on the DF: the DF MUST NOT forward BUM or unicast traffic received from a given split-horizon-group to a member of his own split-horizon group or to the non-DF. Forwarding to the other split-horizon-group (except the non-DF) and local ACs is allowed (as long as the ACs are not part of an ES for which the node is non-DF).

o All-active multi-homing; assuming a WAN split-horizon-group, a DC split-horizon-group and local ACs on the GWs:

+ Forwarding behavior on the non-DF: the non-DF follows the same behavior as the non-DF in the single-active case but only for BUM traffic. Unicast traffic received from a split-horizon-group MUST NOT be forwarded to a member of its own split-horizon-group but can be forwarded normally to the other split-horizon-group and local ACs. If a known unicast packet is identified as a "flooded" packet, the procedures for BUM traffic MUST be followed.

+ Forwarding behavior on the DF: the DF follows the same behavior as the DF in the single-active case but only for BUM traffic. Unicast traffic received from a split-horizon-group MUST NOT be forwarded to a member of its own split-horizon-group but can be forwarded normally to the other split-horizon-group and local ACs. If a known unicast packet is identified as a "flooded" packet, the procedures for BUM traffic MUST be followed.

o No ESI label is required to be signaled for I-ESI for its use by the non-DF in the data path. This is possible because the non-DF and the DF will never forward BUM traffic (coming from a split-horizon-group) to each other.

3.4.4. Impact on MAC Mobility procedures

Since the MAC/IP Advertisement routes are not reflected in the GWs but rather consumed and re-advertised if active, the MAC Mobility procedures can be constrained to each domain (DC or WAN) and resolved within each domain. In other words, if a MAC moves within the DC, the GW MUST NOT re-advertise the route to the WAN with a change in the sequence number. Only when the MAC moves from the WAN domain to the

DC domain (or from one DC to another) the GW will re-advertise the MAC with a higher sequence number in the MAC Mobility extended community. In respect to the MAC Mobility procedures described in [RFC7432] the MAC addresses learned from the NVEs in the local DC or on the local ACs will be considered as local.

The sequence numbers MUST NOT be propagated between domains. The sticky bit indication in the MAC Mobility extended community MUST be propagated between domains.

3.4.5. Gateway optimizations

All the Gateway optimizations described in section 2.5 MAY be applied to the GWs when the Interconnect is based on EVPN-MPLS.

In particular, the use of the Unknown MAC route, as described in section 2.5.1, reduces the unknown flooding in the DC but also solves some transient packet duplication issues in cases of all-active multi-homing. This is explained in the following paragraph.

Consider the diagram in Figure 2 for EVPN-MPLS Interconnect and all-active multi-homing, and the following sequence:

a) MAC Address M1 is advertised from NVE3 in EVI-1.

b) GW3 and GW4 learn M1 for EVI-1 and re-advertise M1 to the WAN with I-ESI-2 in the ESI field.

c) GW1 and GW2 learn M1 and install GW3/GW4 as next-hops following the EVPN aliasing procedures.

d) Before NVE1 learns M1, a packet arrives to NVE1 with destination M1. The packet is subsequently flooded.

e) Since both GW1 and GW2 know M1, they both forward the packet to the WAN (hence creating packet duplication), unless there is an indication in the data plane that the packet from NVE1 has been flooded. If the GWs signal the same VNI/VSID for MAC/IP advertisement and inclusive multicast routes for EVI-1, such data plane indication does not exist.

This undesired situation can be avoided by the use of the Unknown-MAC-route. If this route is used, the NVEs will prune their unknown unicast flooding list, and the non-DF GW will not received unknown packets, only the DF will. This solves the MAC duplication issue described above.

3.4.6. Benefits of the EVPN-MPLS Interconnect solution

Besides retaining the EVPN attributes between Data Centers and throughout the WAN, the EVPN-MPLS Interconnect solution on the GWs has some benefits compared to pure BGP EVPN RR or Inter-AS model B solutions without a gateway:

- o The solution supports the connectivity of local attachment circuits on the GWs.
- o Different data plane encapsulations can be supported in the DC and the WAN.
- o Optimized multicast solution, with independent inclusive multicast trees in DC and WAN.
- o MPLS Label aggregation: for the case where MPLS labels are signaled from the NVEs for MAC/IP Advertisement routes, this solution provides label aggregation. A remote PE MAY receive a single label per GW MAC-VRF as opposed to a label per NVE/MAC-VRF connected to the GW MAC-VRF. For instance, in Figure 2, PE would receive only one label for all the routes advertised for a given MAC-VRF from GW1, as opposed to a label per NVE/MAC-VRF.
- o The GW will not propagate MAC mobility for the MACs moving within a DC. Mobility intra-DC is solved by all the NVEs in the DC. The MAC Mobility procedures on the GWs are only required in case of mobility across DCs.
- o Proxy-ARP/ND function on the DGWs can be leveraged to reduce ARP/ND flooding in the DC or/and in the WAN.

3.5. PBB-EVPN Interconnect for EVPN-Overlay networks

[PBB-EVPN] is yet another Interconnect option. It requires the use of GWs where I-components and associated B-components are EVI instances.

3.5.1. Control/Data Plane setup procedures on the GWs

EVPN will run independently in both components, the I-component MAC-VRF and B-component MAC-VRF. Compared to [PBB-EVPN], the DC C-MACs are no longer learnt in the data plane on the GW but in the control plane through EVPN running on the I-component. Remote C-MACs coming from remote PEs are still learnt in the data plane. B-MACs in the B-component will be assigned and advertised following the procedures described in [PBB-EVPN].

An I-ESI will be configured on the GWs for multi-homing, but it will only be used in the EVPN control plane for the I-component EVI. No

non-reserved ESIs will be used in the control plane of the B-component EVI as per [PBB-EVPN].

The rest of the control plane procedures will follow [RFC7432] for the I-component EVI and [PBB-EVPN] for the B-component EVI.

From the data plane perspective, the I-component and B-component EVPN bindings established to the same far-end will be compared and the I-component EVPN-overlay binding will be kept down following the rules described in section 3.3.1.

3.5.2. Multi-homing procedures on the GWs

Single-active as well as all-active multi-homing MUST be supported.

The forwarding behavior of the DF and non-DF will be changed based on the description outlined in section 3.4.3, only replacing the "WAN split-horizon-group" for the B-component.

3.5.3. Impact on MAC Mobility procedures

C-MACs learnt from the B-component will be advertised in EVPN within the I-component EVI scope. If the C-MAC was previously known in the I-component database, EVPN would advertise the C-MAC with a higher sequence number, as per [RFC7432]. From a Mobility perspective and the related procedures described in [RFC7432], the C-MACs learnt from the B-component are considered local.

3.5.4. Gateway optimizations

All the considerations explained in section 3.4.5 are applicable to the PBB-EVPN Interconnect option.

3.6. EVPN-VXLAN Interconnect for EVPN-Overlay networks

If EVPN for Overlay tunnels is supported in the WAN and a GW function is required, an end-to-end EVPN solution can be deployed. This section focuses on the specific case of EVPN for VXLAN (EVPN-VXLAN hereafter) and the impact on the [RFC7432] procedures.

This use-case assumes that NVEs need to use the VNIs or VSIDs as a globally unique identifiers within a data center, and a Gateway needs to be employed at the edge of the data center network to translate the VNI or VSID when crossing the network boundaries. This GW function provides VNI and tunnel IP address translation. The use-case in which local downstream assigned VNIs or VSIDs can be used (like MPLS labels) is described by [EVPN-Overlays].

While VNIs are globally significant within each DC, there are two possibilities in the Interconnect network:

a) Globally unique VNIs in the Interconnect network:
In this case, the GWs and PEs in the Interconnect network will agree on a common VNI for a given EVI. The RT to be used in the Interconnect network can be auto-derived from the agreed Interconnect VNI. The VNI used inside each DC MAY be the same as the Interconnect VNI.

b) Downstream assigned VNIs in the Interconnect network.
In this case, the GWs and PEs MUST use the proper RTs to import/export the EVPN routes. Note that even if the VNI is downstream assigned in the Interconnect network, and unlike option B, it only identifies the <Ethernet Tag, GW> pair and not the <Ethernet Tag, egress PE> pair. The VNI used inside each DC MAY be the same as the Interconnect VNI. GWs SHOULD support multiple VNI spaces per EVI (one per Interconnect network they are connected to).

In both options, NVEs inside a DC only have to be aware of a single VNI space, and only GWs will handle the complexity of managing multiple VNI spaces. In addition to VNI translation above, the GWs will provide translation of the tunnel source IP for the packets generated from the NVEs, using their own IP address. GWs will use that IP address as the BGP next-hop in all the EVPN updates to the Interconnect network.

The following sections provide more details about these two options.

3.6.1. Globally unique VNIs in the Interconnect network

Considering Figure 2, if a host H1 in NVO-1 needs to communicate with a host H2 in NVO-2, and assuming that different VNIs are used in each DC for the same EVI, e.g. VNI-10 in NVO-1 and VNI-20 in NVO-2, then the VNIs must be translated to a common Interconnect VNI (e.g. VNI-100) on the GWs. Each GW is provisioned with a VNI translation mapping so that it can translate the VNI in the control plane when sending BGP EVPN route updates to the Interconnect network. In other words, GW1 and GW2 must be configured to map VNI-10 to VNI-100 in the BGP update messages for H1’s MAC route. This mapping is also used to translate the VNI in the data plane in both directions, that is, VNI-10 to VNI-100 when the packet is received from NVO-1 and the reverse mapping from VNI-100 to VNI-10 when the packet is received from the remote NVO-2 network and needs to be forwarded to NVO-1.

The procedures described in section 3.4 will be followed, considering that the VNIs advertised/received by the GWs will be translated

accordingly.

3.6.2. Downstream assigned VNIs in the Interconnect network

In this case, if a host H1 in NVO-1 needs to communicate with a host H2 in NVO-2, and assuming that different VNIs are used in each DC for the same EVI, e.g. VNI-10 in NVO-1 and VNI-20 in NVO-2, then the VNIs must be translated as in section 3.6.1. However, in this case, there is no need to translate to a common Interconnect VNI on the GWs. Each GW can translate the VNI received in an EVPN update to a locally assigned VNI advertised to the Interconnect network. Each GW can use a different Interconnect VNI, hence this VNI does not need to be agreed on all the GWs and PEs of the Interconnect network.

The procedures described in section 3.4 will be followed, taking the considerations above for the VNI translation.

5. Conventions and 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].

AC: Attachment Circuit

BUM: it refers to the Broadcast, Unknown unicast and Multicast traffic

DF: Designated Forwarder

GW: Gateway or Data Center Gateway

DCI: Data Center Interconnect

ES: Ethernet Segment

ESI: Ethernet Segment Identifier

I-ESI: Interconnect ESI defined on the GWs for multi-homing to/from the WAN

EVI: EVPN Instance

MAC-VRF: it refers to an EVI instance in a particular node

NVE: Network Virtualization Edge

PW: Pseudowire

RD: Route-Distinguisher

RT: Route-Target

TOR: Top-Of-Rack switch

VNI/VSID: refers to VXLAN/NVGRE virtual identifiers

VSI: Virtual Switch Instance or VPLS instance in a particular PE

6. Security Considerations

This section will be completed in future versions.

7. IANA Considerations

8. References

8.1. Normative References

[RFC4761]Kompella, K., Ed., and Y. Rekhter, Ed., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, DOI 10.17487/RFC4761, January 2007, <http://www.rfc-editor.org/info/rfc4761>.

[RFC4762]Lasserre, M., Ed., and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <http://www.rfc-editor.org/info/rfc4762>.

[RFC6074]Rosen, E., Davie, B., Radoaca, V., and W. Luo, "Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)", RFC 6074, DOI 10.17487/RFC6074, January 2011, <http://www.rfc-editor.org/info/rfc6074>.

[RFC7041]Balus, F., Ed., Sajassi, A., Ed., and N. Bitar, Ed., "Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging", RFC 7041, DOI 10.17487/RFC7041, November 2013, <http://www.rfc-editor.org/info/rfc7041>.

[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>.

8.2. Informative References

[PBB-EVPN] Sajassi et al., "PBB-EVPN", draft-ietf-l2vpn-pbb-evpn-10, work in progress, May, 2015

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

[EVPN-VPLS-INTEGRATION] Sajassi et al., "(PBB-)EVPN Seamless Integration with (PBB-)VPLS", draft-ietf-bess-evpn-vpls-integration-00.txt, work in progress, February, 2015

9. Acknowledgments

The authors would like to thank Neil Hart for their valuable comments and feedback.

10. Contributors

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

Florin Balus
John Drake

11. Authors' Addresses

Jorge Rabadan
Alcatel-Lucent
777 E. Middlefield Road
Mountain View, CA 94043 USA
Email: jorge.rabadan@alcatel-lucent.com

Senthil Sathappan
Alcatel-Lucent
Email: senthil.sathappan@alcatel-lucent.com

Wim Henderickx
Alcatel-Lucent
Email: wim.henderickx@alcatel-lucent.com

Senad Palislamovic
Alcatel-Lucent
Email: senad.palislamovic@alcatel-lucent.com

Ali Sajassi

Cisco
Email: sajassi@cisco.com

Ravi Shekhar
Juniper
Email: rshekhar@juniper.net

Anil Lohiya
Juniper
Email: alohiya@juniper.net

Dennis Cai
Cisco Systems
Email: dcai@cisco.com

BESS Workgroup
INTERNET-DRAFT
Intended Status: Standards Track
Updates: 7385

A. Sajassi, Ed.
S. Salam
Cisco
J. Drake
Juniper
J. Uttaro
ATT
S. Boutros
VMware
J. Rabadan
Nokia

Expires: April 28, 2018

October 28, 2017

E-TREE Support in EVPN & PBB-EVPN
draft-ietf-bess-evpn-etree-14

Abstract

The Metro Ethernet Forum (MEF) has defined a rooted-multipoint Ethernet service known as Ethernet Tree (E-Tree). A solution framework for supporting this service in MPLS networks is described in RFC7387 ("A Framework for Ethernet-Tree (E-Tree) Service over a Multiprotocol Label Switching (MPLS) Network"). This document discusses how those functional requirements can be met with a solution based on RFC7432, BGP MPLS Based Ethernet VPN (EVPN), with some extensions and how such a solution can offer a more efficient implementation of these functions than that of RFC7796, E-Tree Support in Virtual Private LAN Service (VPLS). This document makes use of the most significant bit of the "Tunnel Type" field (in PMSI Tunnel Attribute) governed by the IANA registry created by RFC7385, and hence updates RFC7385 accordingly.

Status of this Memo

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 Metro Ethernet Forum (MEF) has defined a rooted-multipoint Ethernet service known as Ethernet Tree (E-Tree) [MEF6.1]. In an E-Tree service, a customer site that is typically represented by an Attachment Circuits (AC) (e.g., a 802.1Q VLAN tag but may also be represented by a MAC address) is labeled as either a Root or a Leaf site. Root sites can communicate with all other customer sites (both Root and Leaf sites). However, Leaf sites can communicate with Root sites but not with other Leaf sits. In this document unless explicitly mentioned otherwise, a site is always represented by an AC.

[RFC7387] describes a solution framework for supporting E-Tree service in MPLS networks. The document identifies the functional components of an overall solution to emulate E-Tree services in MPLS networks in addition to multipoint-to-multipoint Ethernet LAN (E-LAN) services specified in [RFC7432] and [RFC7623].

[RFC7432] defines EVPN, a solution for multipoint L2VPN services with advanced multi-homing capabilities, using BGP for distributing customer/client MAC address reach-ability information over the MPLS/IP network. [RFC7623] combines the functionality of EVPN with [802.1ah] Provider Backbone Bridging (PBB) for MAC address scalability.

This document discusses how the functional requirements for E-Tree service can be met with a solution based on (PBB-)EVPN (i.e., [RFC7432] and [RFC7623]) with some extensions to their procedures and BGP attributes. Such (PBB-)EVPN based solution can offer a more efficient implementation of these functions than that of RFC7796, E-Tree Support in Virtual Private LAN Service (VPLS). This efficiency is achieved by performing filtering of unicast traffic at the ingress PE nodes as opposed to egress filtering where the traffic is sent through the network and gets filtered and discarded at the egress PE nodes. The details of this ingress filtering is described in section 3.1. Since this document specifies a solution based on [RFC7432], it requires the readers to have the knowledge of [RFC7432] as prerequisite. This document makes use of the most significant bit of the "Tunnel Type" field (in PMSI Tunnel Attribute) governed by the IANA registry created by RFC7385, and hence updates RFC7385 accordingly. Section 2 discusses E-Tree scenarios. Section 3 and 4 describe E-Tree solutions for EVPN and PBB-EVPN respectively, and section 5 covers BGP encoding for E-Tree solutions.

1.1 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 RFC 2119 [KEYWORDS].

1.2 Terminology

Broadcast Domain: In a bridged network, the broadcast domain corresponds to a Virtual LAN (VLAN), where a VLAN is typically represented by a single VLAN ID (VID) but can be represented by several VIDs where Shared VLAN Learning (SVL) is used per [802.1Q].

Bridge Table: An instantiation of a broadcast domain on a MAC-VRF.

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

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.

Ethernet Segment (ES): When a customer site (device or network) is connected to one or more PEs via a set of Ethernet links, then that set of links is referred to as an 'Ethernet segment'.

Ethernet Segment Identifier (ESI): A unique non-zero identifier that identifies an Ethernet segment is called an 'Ethernet Segment Identifier'.

Ethernet Tag: An Ethernet tag identifies a particular broadcast domain, e.g., a VLAN. An EVPN instance consists of one or more broadcast domains.

P2MP: Point to Multipoint.

PE: Provider Edge device.

2 E-Tree Scenarios

This document categorizes E-Tree scenarios into the following three scenarios, depending on the nature of the Root/Leaf site association:

- Either Leaf or Root site(s) per PE

- Either Leaf or Root site(s) per Attachment Circuit (AC)

- Either Leaf or Root site(s) per MAC address

2.1 Scenario 1: Leaf or Root Site(s) per PE

In this scenario, a PE may receive traffic from either Root ACs or Leaf ACs for a given MAC-VRF/bridge table, but not both. In other words, a given EVPN Instance (EVI) on a Provider Edge (PE) device is either associated with Root(s) or Leaf(s). The PE may have both Root and Leaf ACs albeit for different EVIs.

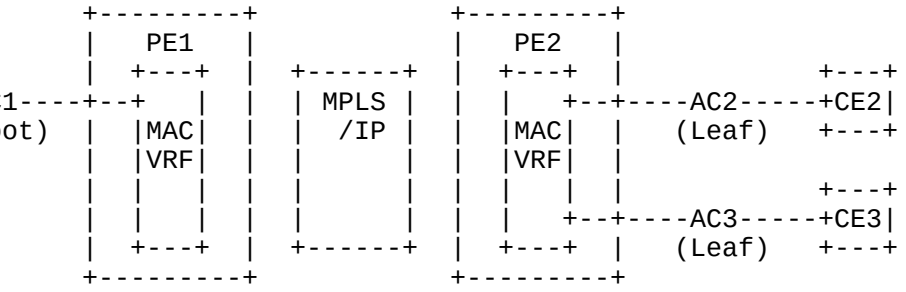


Figure 1: Scenario 1

In this scenario, tailored BGP Route Target (RT) import/export policies among the PEs belonging to the same EVI can be used to prevent the communications among Leaf PEs. To prevent the communications among Leaf ACs connected to the same PE and belonging to the same EVI, split-horizon filtering is used to block traffic from one Leaf AC to another Leaf AC on a MAC-VRF for a given E-Tree EVI. The purpose of this topology constraint is to avoid having PEs with only Leaf sites importing and processing BGP MAC routes from each other. To support such topology constrain in EVPN, two BGP Route-Targets (RTs) are used for every EVPN Instance (EVI): one RT is associated with the Root sites (Root ACs) and the other is associated with the Leaf sites (Leaf ACs). On a per EVI basis, every PE exports the single RT associated with its type of site(s). Furthermore, a PE with Root site(s) imports both Root and Leaf RTs, whereas a PE with Leaf site(s) only imports the Root RT.

For this scenario, if it is desired to use only a single RT per EVI (just like E-LAN services in [RFC7432]), then the approach B in scenario 2 (described below) needs to be used.

2.2 Scenario 2: Leaf or Root Site(s) per AC

In this scenario, a PE can receive traffic from both Root ACs and Leaf ACs for a given EVI. In other words, a given EVI on a PE can be associated with both Root(s) and Leaf(s).

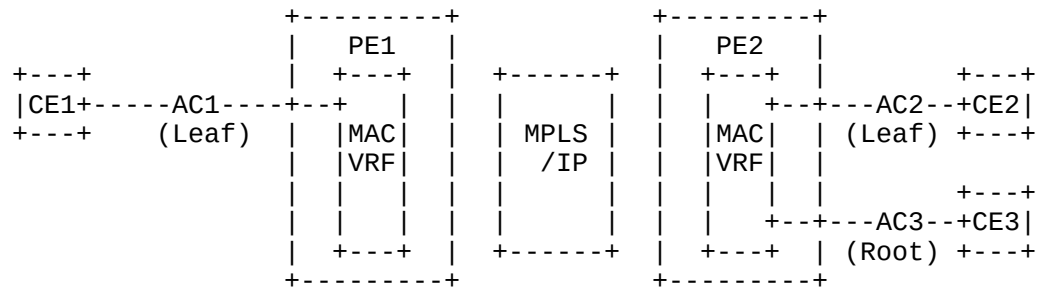


Figure 2: Scenario 2

In this scenario, just like the previous scenario (in section 2.1), two Route Targets (one for Root and another for Leaf) can be used. However, the difference is that on a PE with both Root and Leaf ACs, all remote MAC routes are imported and thus there needs to be a way to differentiate remote MAC routes associated with Leaf ACs versus the ones associated with Root ACs in order to apply the proper ingress filtering.

In order to recognize the association of a destination MAC address to a Leaf or Root AC and thus support ingress filtering on the ingress PE with both Leaf and Root ACs, MAC addresses need to be colored with Root or Leaf indication before advertisements to other PEs. There are two approaches for such coloring:

A) To always use two RTs (one to designate Leaf RT and another for Root RT)

B) To allow for a single RT be used per EVI just like [RFC7432] and thus color MAC addresses via a "color" flag in a new extended community as detailed in section 5.1.

Approach (A) would require the same data plane enhancements as approach (B) if MAC-VRF and bridge tables used per VLAN, are to remain consistent with [RFC7432] (section 6). In order to avoid data-plane enhancements for approach (A), multiple bridge tables per VLAN may be considered; however, this has major drawbacks as described in appendix-A and thus is not recommended.

Given that both approaches (A) and (B) would require the same data-plane enhancements, approach (B) is chosen here in order to allow for RT usage consistent with baseline EVPN [RFC7432] and for better generality. It should be noted that if one wants to use RT constraints in order to avoid MAC advertisements associated with a Leaf AC to PEs with only Leaf ACs, then two RTs (one for Root and another for Leaf) can still be used with approach (B); however, in

such applications Leaf/Root RTs will be used to constrain MAC advertisements and they are not used to color the MAC routes for ingress filtering - i.e., in approach (B), the coloring is always done via the new extended community.

If, for a given EVI, a significant number of PEs have both Leaf and Root sites attached (even though they may start as Root-only or Leaf-only PEs), then a single RT per EVI should be used. The reason for such recommendation is to alleviate the configuration overhead associated with using two RTs per EVI at the expense of having some unwanted MAC addresses on the Leaf-only PEs.

2.3 Scenario 3: Leaf or Root Site(s) per MAC Address

In this scenario, a customer Root or Leaf site is represented by a MAC address and a PE may receive traffic from both Root AND Leaf sites on a single Attachment Circuit (AC) of an EVI. This scenario is not covered in either [RFC7387] or [MEF6.1]; however, it is covered in this document for the sake of completeness. In this scenario, since an AC carries traffic from both Root and Leaf sites, the granularity at which Root or Leaf sites are identified is on a per MAC address. This scenario is considered in this document for EVPN service with only known unicast traffic because the Designated Forwarding (DF) filtering per [RFC7432] would not be compatible with the required egress filtering - i.e., Broadcast, Unknown, and Multicast (BUM) traffic is not supported in this scenario and it is dropped by the ingress PE.

For this scenario, the approach B in scenario 2 (described above) is used in order to allow for single RT usage by service providers.

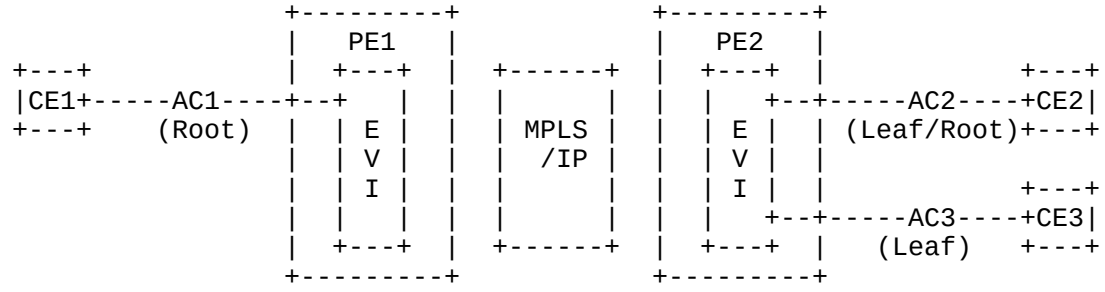


Figure 3: Scenario 3

In conclusion, the approach B in scenario 2 is the recommended approach across all the above three scenarios and the corresponding solution is detailed in the following sections.

3 Operation for EVPN

[RFC7432] defines the notion of Ethernet Segment Identifier (ESI) MPLS label used for split-horizon filtering of BUM traffic at the egress PE. Such egress filtering capabilities can be leveraged in provision of E-Tree services as it will be seen shortly for BUM traffic. For know unicast traffic, additional extensions to [RFC7432] is needed (i.e., a new BGP Extended Community for Leaf indication described in section 5.1) in order to enable ingress filtering as described in detail in the following sections.

3.1 Known Unicast Traffic

Since in EVPN, MAC learning is performed in the control plane via advertisement of BGP routes, the filtering needed by E-Tree service for known unicast traffic can be performed at the ingress PE, thus providing very efficient filtering and avoiding sending known unicast traffic over the MPLS/IP core to be filtered at the egress PE as done in traditional E-Tree solutions - i.e., E-Tree for VPLS [RFC7796].

To provide such ingress filtering for known unicast traffic, a PE MUST indicate to other PEs what kind of sites (Root or Leaf) its MAC addresses are associated with. This is done by advertising a Leaf indication flag (via an Extended Community) along with each of its MAC/IP Advertisement routes learned from a Leaf site. The lack of such flag indicates that the MAC address is associated with a Root site. This scheme applies to all scenarios described in section 2.

Tagging MAC addresses with a Leaf indication enables remote PEs to perform ingress filtering for known unicast traffic - i.e., on the ingress PE, the MAC destination address lookup yields, in addition to the forwarding adjacency, a flag which indicates whether the target MAC is associated with a Leaf site or not. The ingress PE cross-checks this flag with the status of the originating AC, and if both are leafs, then the packet is not forwarded.

In situation where MAC moves are allowed among Leaf and Root sites (e.g., non-static MAC), PEs can receive multiple MAC/IP advertisements routes for the same MAC address with different Leaf/Root indications (and possibly different ESIs for multi-homing scenarios). In such situations, MAC mobility procedures (section 15 of [RFC7432]) take precedence to first identify the location of the MAC before associating that MAC with a Root or a Leaf site.

To support the above ingress filtering functionality, a new E-Tree Extended Community with a Leaf indication flag is introduced [section 5.1]. This new Extended Community MUST be advertised with MAC/IP Advertisement routes learned from a Leaf site. Besides MAC/IP Advertisement route, no other EVPN routes are required to carry this new extended community.

3.2 Broadcast, Unkonwn, and Multicast (BUM) Traffic

This specification does not provide support for filtering BUM (Broadcast, Unknown, and Multicast) traffic on the ingress PE; due to the multi-destination nature of BUM traffic, is is not possible to perform filtering of the same on the ingress PE. As such, the solution relies on egress filtering. In order to apply the proper egress filtering, which varies based on whether a packet is sent from a Leaf AC or a Root AC, the MPLS-encapsulated frames MUST be tagged with an indication when they originated from a Leaf AC - i.e., to be tagged with a Leaf label as specified in section 5.1. This Leaf label allows for disposition PE (e.g., egress PE) to perform the necessary egress filtering function in data-plane similar to ESI label in [RFC7432]. The allocation of the Leaf label is on a per PE basis (e.g., independent of ESI and EVI) as descried in the following sections.

The Leaf label can be upstream assigned for P2MP LSP or downstream assigned for ingress replication tunnels. The main difference between downstream and upstream assigned Leaf label is that in case of downstream assigned not all egress PE devices need to receive the label in MPLS encapsulated BUM packets just like ESI label for ingress replication procedures defined in [RFC7432].

On the ingress PE, the PE needs to place all its Leaf ACs for a given bridge domain in a single split-horizon group in order to prevent intra-PE forwarding among its Leaf ACs. This intra-PE split-horizon filtering applies to BUM traffic as well as known-unicast traffic.

There are four scenarios to consider as follows. In all these scenarios, the ingress PE imposes the right MPLS label associated with the originated Ethernet Segment (ES) depending on whether the Ethernet frame originated from a Root or a Leaf site on that Ethernet Segment (ESI label or Leaf label). The mechanism by which the PE identifies whether a given frame originated from a Root or a Leaf site on the segment is based on the AC identifier for that segment (e.g., Ethernet Tag of the frame for 802.1Q frames). Other mechanisms for identifying Root or Leaf sites such as the use of source MAC address of the receiving frame are optional. The scenarios below are described in context of Root/Leaf AC; however, they can be extended to Root/Leaf MAC address if needed.

3.2.1 BUM Traffic Originated from a Single-homed Site on a Leaf AC

In this scenario, the ingress PE adds a Leaf label advertised using the E-Tree Extended Community (Section 5.1) indicating a Leaf site. This Leaf label, used for single-homing scenarios, is not on a per ES basis but rather on a per PE basis - i.e., a single Leaf MPLS label is used for all single-homed ES's on that PE. This Leaf label is advertised to other PE devices, using the E-Tree Extended Community (section 5.1) along with an Ethernet Auto-discovery per ES (EAD-ES) route with ESI of zero and a set of Route Targets (RTs) corresponding to all EVIs on the PE where each EVI has at least one Leaf site. Multiple EAD-ES routes will need to be advertised if the number of Route Targets (RTs) that need to be carried exceed the limit on a single route per [RFC7432]. The ESI for the EAD-ES route is set to zero to indicate single-homed sites.

When a PE receives this special Leaf label in the data path, it blocks the packet if the destination AC is of type Leaf; otherwise, it forwards the packet.

3.2.2 BUM Traffic Originated from a Single-homed Site on a Root AC

In this scenario, the ingress PE does not add any ESI label or Leaf label and it operates per [RFC7432] procedures.

3.2.3 BUM Traffic Originated from a Multi-homed Site on a Leaf AC

In this scenario, it is assumed that while different ACs (VLANs) on the same ES could have different Root/Leaf designation (some being Roots and some being Leafs), the same VLAN does have the same Root/Leaf designation on all PEs on the same ES. Furthermore, it is assumed that there is no forwarding among subnets - ie, the service is EVPN L2 and not EVPN IRB [EVPN-IRB]. IRB use cases described in [EVPN-IRB] are outside the scope of this document.

In this scenario, if a multicast or broadcast packet is originated from a Leaf AC, then it only needs to carry Leaf label described in section 3.2.1. This label is sufficient in providing the necessary egress filtering of BUM traffic from getting sent to Leaf ACs including the Leaf AC on the same Ethernet Segment.

3.2.4 BUM Traffic Originated from a Multi-homed Site on a Root AC

In this scenario, both the ingress and egress PE devices follows the procedure defined in [RFC7432] for adding and/or processing an ESI MPLS label - i.e., existing procedures for BUM traffic in [RFC7432] are sufficient and there is no need to add a Leaf label.

3.3 E-Tree Traffic Flows for EVPN

Per [RFC7387], a generic E-Tree service supports all of the following traffic flows:

- Known unicast traffic from Root to Roots & Leaf
- Known unicast traffic from Leaf to Root
- BUM traffic from Root to Roots & Leafs
- BUM traffic from Leaf to Roots

A particular E-Tree service may need to support all of the above types of flows or only a select subset, depending on the target application. In the case where only multicast and broadcast flows need to be supported, the L2VPN PEs can avoid performing any MAC learning function.

The following subsections will describe the operation of EVPN to support E-Tree service with and without MAC learning.

3.3.1 E-Tree with MAC Learning

The PEs implementing an E-Tree service must perform MAC learning when unicast traffic flows must be supported among Root and Leaf sites. In this case, the PE(s) with Root sites performs MAC learning in the data-path over the Ethernet Segments, and advertises reachability in EVPN MAC/IP Advertisement Routes. These routes will be imported by all PEs for that EVI (i.e., PEs that have Leaf sites as well as PEs that have Root sites). Similarly, the PEs with Leaf sites perform MAC learning in the data-path over their Ethernet Segments, and advertise reachability in EVPN MAC/IP Advertisement Routes. For scenarios where two different RTs are used per EVI (one to designate Root site and another to designate Leaf site), the MAC/IP Advertisement routes are imported only by PEs with at least one Root site in the EVI - i.e., a PE with only Leaf sites will not import these routes. PEs with Root and/or Leaf sites may use the Ethernet Auto-discovery per EVI (EAD-EVI) routes for aliasing (in the case of multi-homed segments) and EAD-ES routes for mass MAC withdrawal per [RFC7432].

To support multicast/broadcast from Root to Leaf sites, either a P2MP tree rooted at the PE(s) with the Root site(s) (e.g., Root PEs) or ingress replication can be used (section 16 of [RFC7432]). The multicast tunnels are set up through the exchange of the EVPN Inclusive Multicast route, as defined in [RFC7432].

To support multicast/broadcast from Leaf to Root sites, either ingress replication tunnels from each Leaf PE or a P2MP tree rooted at each Leaf PE can be used. The following two paragraphs describes

when each of these tunneling schemes can be used and how to signal them.

When there are only a few Root PEs with small amount of multicast/broadcast traffic from Leaf PEs toward Root PEs, then ingress replication tunnels from Leaf PEs toward Root PEs should be sufficient. Therefore, if a Root PE needs to support a P2MP tunnel in transmit direction from itself to Leaf PEs and at the same time it wants to support ingress-replication tunnels in receive direction, the Root PE can signal it efficiently by using a new composite tunnel type defined in section 5.2. This new composite tunnel type is advertised by the Root PE to simultaneously indicate a P2MP tunnel in transmit direction and an ingress-replication tunnel in the receive direction for the BUM traffic.

If the number of Root PEs is large, P2MP tunnels (e.g., mLDP or RSVP-TE) originated at the Leaf PEs may be used and thus there will be no need to use the modified PMSI tunnel attribute and the composite tunnel type values defined in section 5.2.

3.3.2 E-Tree without MAC Learning

The PEs implementing an E-Tree service need not perform MAC learning when the traffic flows between Root and Leaf sites are mainly multicast or broadcast. In this case, the PEs do not exchange EVPN MAC/IP Advertisement Routes. Instead, the Inclusive Multicast Ethernet Tag route is used to support BUM traffic. In such scenarios, the small amount of unicast traffic (if any) is sent as part of BUM traffic.

The fields of this route are populated per the procedures defined in [RFC7432], and the multicast tunnel setup criteria are as described in the previous section.

Just as in the previous section, if the number of Root PEs are only a few and thus ingress replication is desired from Leaf PEs to these Root PEs, then the modified PMSI attribute and the composite tunnel type values defined in section 5.2 should be used.

4 Operation for PBB-EVPN

In PBB-EVPN, the PE advertises a Root/Leaf indication along with each B-MAC Advertisement route to indicate whether the associated B-MAC address corresponds to a Root or a Leaf site. Just like the EVPN case, the new E-Tree Extended Community defined in section [5.1] is advertised with each EVPN MAC/IP Advertisement route.

In the case where a multi-homed Ethernet Segment has both Root and Leaf sites attached, two B-MAC addresses are advertised: one B-MAC address is per ES as specified in [RFC7623] and implicitly denoting Root, and the other B-MAC address is per PE and explicitly denoting Leaf. The former B-MAC address is not advertised with the E-Tree extended community but the latter B-MAC denoting Leaf is advertised with the new E-Tree extended community where "Leaf-indication" flag is set. In multi-homing scenarios where an Ethernet Segment has both Root and Leaf ACs, it is assumed that while different ACs (VLANs) on the same ES could have different Root/Leaf designation (some being Roots and some being Leafs), the same VLAN does have the same Root/Leaf designation on all PEs on the same ES. Furthermore, it is assumed that there is no forwarding among subnets - ie, the service is L2 and not IRB. IRB use case is outside the scope of this document.

The ingress PE uses the right B-MAC source address depending on whether the Ethernet frame originated from the Root or Leaf AC on that Ethernet Segment. The mechanism by which the PE identifies whether a given frame originated from a Root or Leaf site on the segment is based on the Ethernet Tag associated with the frame. Other mechanisms of identification, beyond the Ethernet Tag, are outside the scope of this document.

Furthermore, a PE advertises two special global B-MAC addresses: one for Root and another for Leaf, and tags the Leaf one as such in the MAC Advertisement route. These B-MAC addresses are used as source addresses for traffic originating from single-homed segments. The B-MAC address used for indicating Leaf sites can be the same for both single-homed and multi-homed segments.

4.1 Known Unicast Traffic

For known unicast traffic, the PEs perform ingress filtering: On the ingress PE, the C-MAC [RFC7623] destination address lookup yields, in addition to the target B-MAC address and forwarding adjacency, a flag which indicates whether the target B-MAC is associated with a Root or a Leaf site. The ingress PE also checks the status of the originating site, and if both are a Leaf, then the packet is not forwarded.

4.2 Broadcast, Unkonwn, and Multicast (BUM) Traffic

For BUM traffic, the PEs must perform egress filtering. When a PE receives an EVPN MAC/IP advertisement route (which will be used as a source B-MAC for BUM traffic), it updates its egress filtering (based on the source B-MAC address), as follows:

- If the EVPN MAC/IP Advertisement route indicates that the advertised B-MAC is a Leaf, and the local Ethernet Segment is a Leaf as well, then the source B-MAC address is added to its B-MAC list used for egress filtering - i.e., to block traffic from that B-MAC address.

- Otherwise, the B-MAC filtering list is not updated.

- If the EVPN MAC/IP Advertisement route indicates that the advertised B-MAC has changed its designation from a Leaf to a Root and the local Ethernet Segment is a Leaf, then the source B-MAC address is removed from the B-MAC list corresponding to the local Ethernet Segment used for egress filtering - i.e., to unblock traffic from that B-MAC address.

When the egress PE receives the packet, it examines the B-MAC source address to check whether it should filter or forward the frame. Note that this uses the same filtering logic as baseline [RFC7623] for an ESI and does not require any additional flags in the data-plane.

Just as in section 3.2, the PE places all Leaf Ethernet Segments of a given bridge domain in a single split-horizon group in order to prevent intra-PE forwarding among Leaf segments. This split-horizon function applies to BUM traffic as well as known-unicast traffic.

4.3 E-Tree without MAC Learning

In scenarios where the traffic of interest is only multicast and/or broadcast, the PEs implementing an E-Tree service do not need to do any MAC learning. In such scenarios the filtering must be performed on egress PEs. For PBB-EVPN, the handling of such traffic is per section 4.2 without the need for C-MAC learning (in data-plane) in I-component (C-bridge table) of PBB-EVPN PEs (at both ingress and egress PEs).

5 BGP Encoding

This document defines a new BGP Extended Community for EVPN.

5.1 E-Tree Extended Community

This Extended Community is a new transitive Extended Community [RFC4360] having a Type field value of 0x06 (EVPN) and the Sub-Type 0x05. It is used for Leaf indication of known unicast and BUM traffic. It indicates that the frame is originated from a Leaf site.

The E-Tree Extended Community is encoded as an 8-octet value as follows:

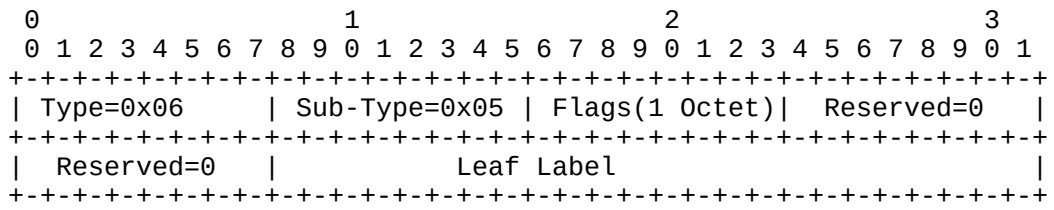
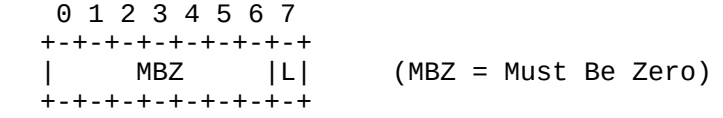


Figure 4: E-Tree Extended Community

The Flags field has the following format:



This document defines the following flags:

+ Leaf-Indication (L)

A value of one indicates a Leaf AC/Site. The rest of flag bits are reserved and should be set to zero.

When this Extended Community (EC) is advertised along with MAC/IP Advertisement route (for known unicast traffic) per section 3.1, the Leaf-Indication flag MUST be set to one and Leaf Label SHOULD be set to zero. The receiving PE MUST ignore Leaf Label and only processes Leaf-Indication flag. A value of zero for Leaf-Indication flag is invalid when sent along with MAC/IP advertisement route and an error should be logged.

When this EC is advertised along with EAD-ES route (with ESI of zero) for BUM traffic to enable egress filtering on disposition PEs per sections 3.2.1 and 3.2.3, the Leaf Label MUST be set to a valid MPLS label (i.e., non-reserved assigned MPLS label [RFC3032]) and the Leaf-Indication flag SHOULD be set to zero. The value of the 20-bit MPLS label is encoded in the high-order 20 bits of the Leaf Label field. The receiving PE MUST ignore the Leaf-Indication flag. A non-valid MPLS label when sent along with the EAD-ES route, should be ignored and logged as an error.

The reserved bits SHOULD be set to zero by the transmitter and MUST

be ignored by the receiver.

5.2 PMSI Tunnel Attribute

[RFC6514] defines PMSI Tunnel attribute which is an optional transitive attribute with the following format:

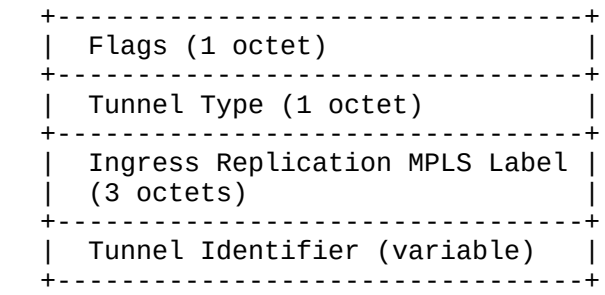


Figure 5: PMSI Tunnel Attribute

This document defines a new Composite tunnel type by introducing a new 'Composite Tunnel' bit in the Tunnel Type field and adding a MPLS label to the Tunnel Identifier field of PMSI Tunnel attribute as detailed below. All other fields remain as defined in [RFC6514]. Composite tunnel type is advertised by the Root PE to simultaneously indicate a non-(ingress replication) tunnel (e.g., P2MP tunnel) in transmit direction and an ingress-replication tunnel in the receive direction for the BUM traffic.

When receiver ingress-replication labels are needed, the high-order bit of the tunnel type field (Composite Tunnel bit) is set while the remaining low-order seven bits indicate the tunnel type as before (for the existing tunnel types). When this Composite Tunnel bit is set, the "tunnel identifier" field begins with a three-octet label, followed by the actual tunnel identifier for the transmit tunnel. PEs that don't understand the new meaning of the high-order bit treat the tunnel type as an undefined tunnel type and treat the PMSI tunnel attribute as a malformed attribute [RFC6514]. That is why the composite tunnel bit is allocated in the Tunnel Type field rather than the Flags field. For the PEs that do understand the new meaning of the high-order, if ingress replication is desired when sending BUM traffic, the PE will use the the label in the Tunnel Identifier field when sending its BUM traffic.

Using the Composite Tunnel bit for Tunnel Types 0x00 'no tunnel information present' and 0x06 'Ingress Replication' is invalid, and a

PE that receives a PMSI Tunnel attribute with such information, considers it as malformed and it SHOULD treat this Update as though all the routes contained in this Update had been withdrawn per section 5 of [RFC6514].

6 Acknowledgement

We would like to thank Eric Rosen, Jeffrey Zhang, Wen Lin, Aldrin Issac, Wim Henderickx, Dennis Cai, and Antoni Przygienda for their valuable comments and contributions. The authors would also like to thank Thomas Morin for shepherding this document and providing valuable comments.

7 Security Considerations

Since this document uses the EVPN constructs of [RFC7432] and [RFC7623], the same security considerations in these documents are also applicable here. Furthermore, this document provides an additional security check by allowing sites (or ACs) of an EVPN instance to be designated as "Root" or "Leaf" by the network operator/ service provider and thus preventing any traffic exchange among "Leaf" sites of that VPN through ingress filtering for known unicast traffic and egress filtering for BUM traffic. Since by default and for the purpose of backward compatibility, an AC that doesn’t have a Leaf designation is considered as a Root AC, in order to avoid any traffic exchange among Leaf ACs, the operator SHOULD configure the AC with a proper role (Leaf or Root) before activating the AC.

8 IANA Considerations

IANA has allocated value 5 in the "EVPN Extended Community Sub-Types" registry defined in [RFC7153] as follow:

SUB-TYPE VALUE	NAME	Reference
0x05	E-Tree Extended Community	This document

This document creates a one-octet registry called "E-Tree Flags". New registrations will be made through the "RFC Required" procedure defined in [RFC8126]. Initial registrations are as follows:

bit	Name	Reference
0-6	Unassigned	
7	Leaf-Indication	This document

8.1 Considerations for PMSI Tunnel Types

The "P-Multicast Service Interface Tunnel (PMSI Tunnel) Tunnel Types" registry in the "Border Gateway Protocol (BGP) Parameters" registry needs to be updated to reflect the use of the most significant bit as "Composite Tunnel" bit (section 5.2).

For this purpose, this document updates [RFC7385] by changing the previously unassigned values (i.e., 0x08 - 0xFA) as follow:

Value	Meaning	Reference
0x08-0x7A	Unassigned	
0x7B-0x7E	Experimental	this document
0x7F	Reserved	this document
0x80-0xFA	Reserved for Composite tunnel	this document
0xFB-0xFE	Experimental	[RFC7385]
0xFF	Reserved	[RFC7385]

The allocation policy for values 0x08-0x7A is per IETF Review [RFC8126]. The range for experimental has been expanded to include the previously assigned range of 0xFB-0xFE and the new range of 0x7B-0x7E. The value in these ranges are not to be assigned. The value 0x7F which is the mirror image of (0xFF) is reserved in this document.

9 References

9.1 Normative References

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC8126] Cotton et al, "Guidelines for Writing an IANA Considerations Section in RFCs", June, 2017.

[RFC7387] Key et al., "A Framework for E-Tree Service over MPLS Network", October 2014.

[MEF6.1] Metro Ethernet Forum, "Ethernet Services Definitions - Phase

2", MEF 6.1, April 2008, https://mef.net/PDF_Documents/technical-specifications/MEF6-1.pdf

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

[RFC7623] Sajassi et al., "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", September, 2015.

[RFC7385] Andersson et al., "IANA Registry for P-Multicast Service Interface (PMSI) Tunnel Type Code Points", October, 2014.

[RFC7153] Rosen et al., "IANA Registries for BGP Extended Communities", March, 2014.

[RFC6514] Aggarwal et al., "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", February, 2012.

[RFC4360] Sangli et al., "BGP Extended Communities Attribute", February, 2006.

9.2 Informative References

[RFC4360] S. Sangli et al, "BGP Extended Communities Attribute", February, 2006.

[RFC3032] E. Rosen et al, "MPLS Label Stack Encoding", January 2001.

[RFC7796] Y. Jiang et al, "Ethernet-Tree (E-Tree) Support in Virtual Private LAN Service (VPLS)", March 2016.

[EVPN-IRB] A. Sajassi et al, "Integrated Routing and Bridging in EVPN", draft-ietf-bess-evpn-inter-subnet-forwarding-03, February 8, 2017.

[802.1ah] IEEE, "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", Clauses 25 and 26, IEEE Std 802.1Q, DOI 10.1109/IEEESTD.2011.6009146.

Appendix-A

When two MAC-VRFs (two bridge tables per VLANs) are used for an E-Tree service (one for Root ACs and another for Leaf ACs) on a given PE, then the following complications in data-plane path can result.

Maintaining two MAC-VRFs (two bridge tables) per VLAN (when both Leaf and Root ACs exists for that VLAN) would either require two lookups

be performed per MAC address in each direction in case of a miss, or duplicating many MAC addresses between the two bridge tables belonging to the same VLAN (same E-Tree instance). Unless two lookups are made, duplication of MAC addresses would be needed for both locally learned and remotely learned MAC addresses. Locally learned MAC addresses from Leaf ACs need to be duplicated onto Root bridge table and locally learned MAC addresses from Root ACs need to be duplicated onto Leaf bridge table. Remotely learned MAC addresses from Root ACs need to be copied onto both Root and Leaf bridge tables. Because of potential inefficiencies associated with data-plane implementation of additional MAC lookup or duplication of MAC entries, this option is not believed to be implementable without dataplane performance inefficiencies in some platforms and thus this document introduces the coloring as described in section 2.2 and detailed in section 3.1.

Authors' Addresses

Ali Sajassi
Cisco
Email: sajassi@cisco.com

Samer Salam
Cisco
Email: ssalam@cisco.com

John Drake
Juniper
Email: jdrake@juniper.net

Jim Uttaro
AT&T
Email: ju1738@att.com

Sami Boutros
VMware
Email: sboutros@vmware.com

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

INTERNET-DRAFT
Intended Status: Standard Track

Sami Boutros
VMware
Ali Sajassi
Samer Salam
Cisco Systems
John Drake
Juniper Networks
J. Rabadan
Nokia

Expires: November 15, 2017 May 14, 2017

Virtual Private Wire Service support in Ethernet VPN
draft-ietf-bess-evpn-vpws-14.txt

Abstract

This document describes how Ethernet VPN (EVPN) can be used to support Virtual Private Wire Service (VPWS) in MPLS/IP networks. EVPN enables the following characteristics for VPWS: single-active as well as all-active multi-homing with flow-based load-balancing, eliminates the need for Pseudowire (PW) signaling, and provides fast protection convergence upon node or link failure.

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 describes how EVPN can be used to support VPWS in MPLS/IP networks. The use of EVPN mechanisms for VPWS (EVPN-VPWS) brings the benefits of EVPN to Point to Point (P2P) services. These benefits include single-active redundancy as well as all-active redundancy with flow-based load-balancing. Furthermore, the use of EVPN for VPWS eliminates the need for traditional way of PW signaling for P2P Ethernet services, as described in section 4.

[RFC7432] provides the ability to forward customer traffic to/from a given customer Attachment Circuit (AC), without any Media Access Control (MAC) lookup. This capability is ideal in providing P2P services (aka VPWS services). [MEF] defines Ethernet Virtual Private Line (EVPL) service as P2P service between a pair of ACs (designated by VLANs) and Ethernet Private Line (EPL) service, in which all traffic flows are between a single pair of ports, that in EVPN terminology would mean a single pair of Ethernet Segments ES(es). EVPL can be considered as a VPWS with only two ACs. In delivering an EVPL service, the traffic forwarding capability of EVPN is based on the exchange of a pair of Ethernet Auto-discovery (A-D) routes; whereas, for more general VPWS as per [RFC4664], traffic forwarding capability of EVPN is based on the exchange of a group of Ethernet AD routes (one Ethernet AD route per AC/ES). In a VPWS service, the traffic from an originating Ethernet Segment can be forwarded only to a single destination Ethernet Segment; hence, no MAC lookup is needed and the MPLS label associated with the per EVPN instance (EVI) Ethernet A-D route can be used in forwarding user traffic to the destination AC.

For both EPL and EVPL services, a specific VPWS service instance is identified by a pair of per-EVI Ethernet A-D routes which together identify the VPWS service instance endpoints and the VPWS service instance. In the control plane the VPWS service instance is identified using the VPWS service instance identifiers advertised by each Provider Edge node (PE). In the data plane the value of the MPLS label advertised by one PE is used by the other PE to send traffic for that VPWS service instance. As with the Ethernet Tag in standard EVPN, the VPWS service instance identifier has uniqueness within an EVPN instance.

For EVPN routes, the Ethernet Tag IDs are set to zero for Port-based, VLAN-based, and VLAN-bundle interface mode and set to non-zero Ethernet Tag IDs for VLAN-aware bundle mode. Conversely, for EVPN-VPWS, the Ethernet Tag ID in the Ethernet A-D route MUST be set to a non-zero value for all four service interface types.

In terms of route advertisement and MPLS label lookup behavior, EVPN-

VPWS resembles the VLAN-aware bundle mode of [RFC7432] such that when a PE advertises per-EVI Ethernet A-D route, the VPWS service instance serves as a 32-bit normalized Ethernet Tag ID. The value of the MPLS label in this route represents both the EVI and the VPWS service instance, so that upon receiving an MPLS encapsulated packet, the disposition PE can identify the egress AC from the MPLS label and subsequently perform any required tag translation. For EVPL service, the Ethernet frames transported over an MPLS/IP network SHOULD remain tagged with the originating VLAN-ID (VID) and any VID translation MUST be performed at the disposition PE. For EPL service, the Ethernet frames are transported as is and the tags are not altered.

The MPLS label value in the Ethernet A-D route can be set to the Virtual Extensible LAN (VXLAN) Network Identifier (VNI) for VXLAN encapsulation as per [RFC7348], and this VNI will have a local scope per PE and may also be equal to the VPWS service instance identifier set in the Ethernet A-D route. When using VXLAN encap, the BGP Encapsulation extended community is included in the Ethernet A-D route as described in [ietf-evpn-overlay]. The VXLAN VNI like the MPLS label that will be set in the tunnel header used to tunnel Ethernet packets from all the service interface types defined in section 2. The EVPN-VPWS techniques defined in this document has no dependency on the tunneling technology.

The Ethernet Segment identifier encoded in the Ethernet A-D per-EVI route is not used to identify the service. However it can be used for flow-based load-balancing and mass withdraw functions as per the [RFC7432] baseline.

As with standard EVPN, the Ethernet A-D per-ES route is used for fast convergence upon link or node failure. The Ethernet Segment route is used for auto-discovery of the PEs attached to a given multi-homed Customer Edge node (CE) and to synchronize state between them.

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].

EVPN: Ethernet VPN

MAC: Media Access Control

MPLS: Multi Protocol Label Switching.

OAM: Operations, Administration and Maintenance.

PE: Provide Edge Node.

ASBR: Autonomous System Border Router

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

EVPL: Ethernet Virtual Private Line.

EPL: Ethernet Private Line.

EP-LAN: Ethernet Private LAN.

EVP-LAN: Ethernet Virtual Private LAN.

S-VLAN: Service VLAN identifier.

C-VLAN: Customer VLAN identifier.

VID: VLAN-ID.

VPWS: Virtual Private Wire Service.

EVI: EVPN Instance.

P2P: Point to Point.

VXLAN: Virtual Extensible LAN.

DF: Designated Forwarder.

L2: Layer 2.

MTU: Maximum Transmission Unit.

eBGP: Exterior Border Gateway Protocol.

iBGP: Internal Border Gateway Protocol.

ES: Ethernet Segment on a PE refers to the link attached to it, this link can be part of a set of links attached to different PEs in multi homed cases, or could be a single link in single homed cases.

ESI: Ethernet Segment Identifier.

Single-Active Mode: When a device or a network is multi-homed to two or more PEs and when only a single PE in such redundancy group can

forward traffic to/from the multi-homed device or network for a given VLAN, then such multi-homing or redundancy is referred to as "Single-Active".

All-Active: When a device is multi-homed to two or more PEs and when all PEs in such redundancy group can forward traffic to/from the multi-homed device for a given VLAN, then such multi-homing or redundancy is referred to as "All-Active".

VPWS Service Instance: It is represented by a pair of EVPN service labels associated with a pair of endpoints. Each label is downstream assigned and advertised by the disposition PE through an Ethernet A-D per-EVI route. The downstream label identifies the endpoint on the disposition PE. A VPWS service instance can be associated with only one VPWS service identifier.

2 Service interface

2.1 VLAN-Based Service Interface

With this service interface, a VPWS instance identifier corresponds to only a single VLAN on a specific interface. Therefore, there is a one-to-one mapping between a VID on this interface and the VPWS service instance identifier. The PE provides the cross-connect functionality between an MPLS LSP identified by the VPWS service instance identifier and a specific <port,VLAN>. If the VLAN is represented by different VIDs on different PEs and different ES(es), (e.g., a different VID per Ethernet segment per PE), then each PE needs to perform VID translation for frames destined to its Ethernet segment. In such scenarios, the Ethernet frames transported over an MPLS/IP network SHOULD remain tagged with the originating VID, and a VID translation MUST be supported in the data path and MUST be performed on the disposition PE.

2.2 VLAN Bundle Service Interface

With this service interface, a VPWS service instance identifier corresponds to multiple VLANs on a specific interface. The PE provides the cross-connect functionality between the MPLS label identified by the VPWS service instance identifier and a group of VLANs on a specific interface. For this service interface, each VLAN is presented by a single VID which means no VLAN translation is allowed. The receiving PE, can direct the traffic based on EVPN label alone to a specific port. The transmitting PE can cross-connect traffic from a group of VLANs on a specific port to the MPLS label. The MPLS-encapsulated frames MUST remain tagged with the originating VID.

2.2.1 Port-Based Service Interface

This service interface is a special case of the VLAN bundle service interface, where all of the VLANs on the port are mapped to the same VPWS service instance identifier. The procedures are identical to those described in Section 2.2.

2.3 VLAN-Aware Bundle Service Interface

Contrary to EVPN, in EVPN-VPWS this service interface maps to a VLAN-based service interface (defined in section 2.1) and thus this service interface is not used in EVPN-VPWS. In other words, if one tries to define data plane and control plane behavior for this service interface, one would realize that it is the same as that of VLAN-based service.

3. BGP Extensions

This document specifies the use of the per-EVI Ethernet A-D route to signal VPWS services. The Ethernet Segment Identifier field is set to the customer ES and the Ethernet Tag ID 32-bit field MUST be set to the VPWS service instance identifier value. The VPWS service instance identifier value MAY be set to a 24-bit value and when a 24-bit value is used, it MUST be right aligned. For both EPL and EVPL services using a given VPWS service instance, the pair of PEs instantiating that VPWS service instance will each advertise a per-EVI Ethernet A-D route with its VPWS service instance identifier and will each be configured with the other PE’s VPWS service instance identifier. When each PE has received the other PE’s per-EVI Ethernet A-D route, the VPWS service instance is instantiated. It should be noted that the same VPWS service instance identifier may be configured on both PEs.

The Route-Target (RT) extended community with which the per-EVI Ethernet A-D route is tagged identifies the EVPN instance in which the VPWS service instance is configured. It is the operator’s choice as to how many and which VPWS service instances are configured in a given EVPN instance. However, a given EVPN instance MUST NOT be configured with both VPWS service instances and standard EVPN multi-point services.

3.1 EVPN Layer 2 attributes extended community

This document defines a new extended community [RFC4360], to be included with per-EVI Ethernet A-D routes. This attribute is mandatory if multihoming is enabled.

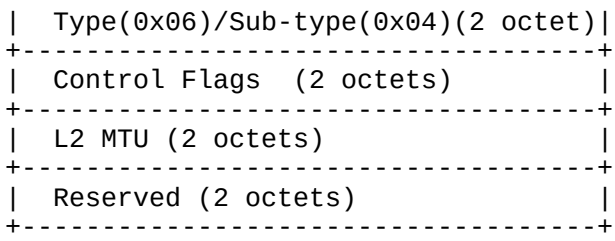


Figure 1: EVPN Layer 2 attributes extended community

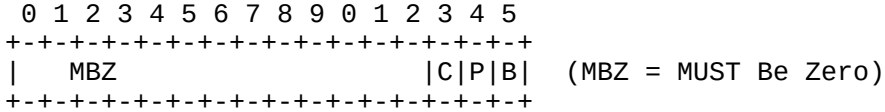


Figure 2: EVPN Layer 2 attributes Control Flags

The following bits in the Control Flags are defined; the remaining bits MUST be set to zero when sending and MUST be ignored when receiving this community.

Name	Meaning
P	If set to 1 in multihoming single-active scenarios, it indicates that the advertising PE is the Primary PE. MUST be set to 1 for multihoming all-active scenarios by all active PE(s).
B	If set to 1 in multihoming single-active scenarios, it indicates that the advertising PE is the Backup PE.
C	If set to 1, a Control word [RFC4448] MUST be present when sending EVPN packets to this PE. It is recommended to include the control word in the absence of Entropy Label.

L2 MTU (Maximum Transmission Unit) is a 2-octet value indicating the MTU in bytes.

A received L2 MTU of zero means no MTU checking against local MTU is needed. A received non-zero MTU MUST be checked against local MTU and if there is a mismatch, the local PE MUST NOT add the remote PE as the EVPN destination for the corresponding VPWS service instance.

The usage of the Per ES Ethernet A-D route is unchanged from its usage in [RFC7432], i.e., the "Single-Active" bit in the flags of the ESI Label extended community will indicate if single-active or all-

active redundancy is used for this ES.

In multihoming scenarios, the B and P flags MUST be cleared. A PE that receives an update with both B and P flags set MUST treat the route as a withdrawal. If the PE receives a route with both B and P clear, it MUST treat the route as a withdrawal from the sender PE.

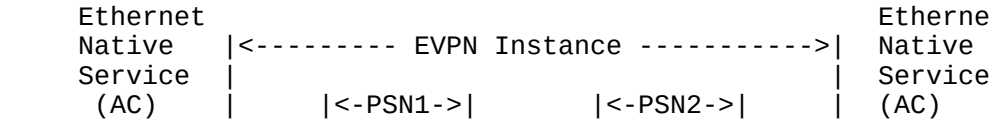
In a multihoming all-active scenario, there is no Designated Forwarder (DF) election, and all the PEs in the ES that are active and ready to forward traffic to/from the CE will set the P Flag. A remote PE will do per-flow load-balancing to the PEs that set the P Flag for the same Ethernet Tag and ESI. The B Flag in control flags SHOULD NOT be set in the multihoming all-active scenario and MUST be ignored by receiving PE(s) if set.

In multihoming single-active scenario for a given VPWS service instance, the DF election should result in the Primary-elected PE for the VPWS service instance advertising the P Flag set and the B Flag clear, the Backup elected PE should advertise the P Flag clear and the B Flag set, and the rest of the PEs in the same ES should signal both P and B Flags clear. When the primary PE/ES fails, the primary PE will withdraw the associated Ethernet A-D routes for the VPWS service instance from the remote PE and the remote PEs should then send traffic associated with the VPWS instance to the backup PE. DF re-election will happen between the PE(s) in the same ES, and there will be a newly elected primary PE and newly elected backup PE that will signal the P and B Flags as described. A remote PE SHOULD receive the P Flag set from only one Primary PE and the B Flag set from only one Backup PE. However during transient situations, a remote PE receiving a P Flag set from more than one PE will select the last advertising PE as the primary PE when forwarding traffic. A remote PE receiving a B Flag set from more than one PE will select the last advertising PE as the backup PE. A remote PE MUST receive P Flag set from at least one PE before forwarding traffic.

If a network uses entropy labels per [RFC6790] then the C Flag MUST NOT be set and control word MUST NOT be used when sending EVPN-encapsulated packets over a P2P LSP.

4 Operation

The following figure shows an example of a P2P service deployed with EVPN.



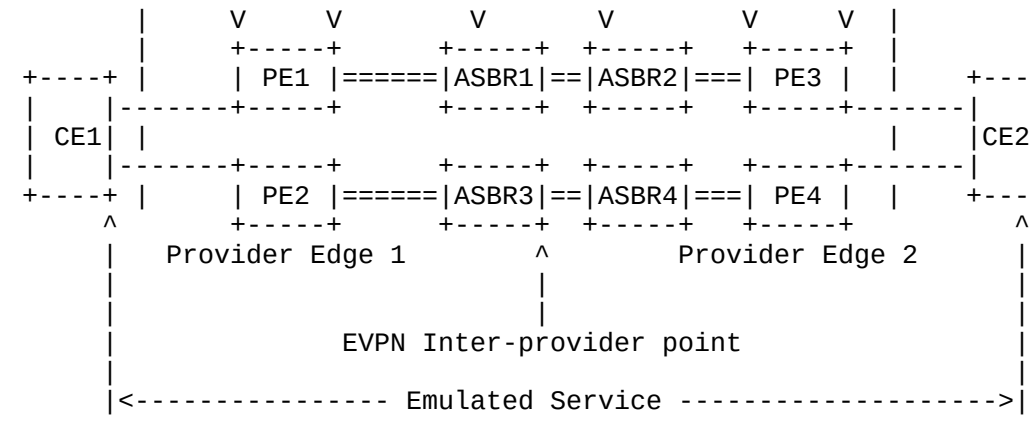


Figure 3: EVPN-VPWS Deployment Model

iBGP sessions are established between PE1, PE2, ASBR1 and ASBR3, possibly via a BGP route-reflector. Similarly, iBGP sessions are established between PE3, PE4, ASBR2 and ASBR4. eBGP sessions are established among ASBR1, ASBR2, ASBR3, and ASBR4.

All PEs and ASBRs are enabled for the EVPN SAFI and exchange per-EVI Ethernet A-D routes, one route per VPWS service instance. For inter-AS option B, the ASBRs re-advertise these routes with the NEXT_HOP attribute set to their IP addresses as per [RFC4271]. The link between the CE and the PE is either a C-tagged or S-tagged interface, as described in [802.1Q], that can carry a single VLAN tag or two nested VLAN tags and it is configured as a trunk with multiple VLANs, one per VPWS service instance. It should be noted that the VLAN ID used by the customer at either end of a VPWS service instance to identify that service instance may be different and EVPN doesn't perform that translation between the two values. Rather, the MPLS label will identify the VPWS service instance and if translation is needed, it should be done by the Ethernet interface for each service.

For single-homed CE, in an advertised per-EVI Ethernet A-D route the ESI field is set to 0 and the Ethernet Tag ID is set to the VPWS service instance identifier that identifies the EVPL or EPL service.

For a multi-homed CE, in an advertised per-EVI Ethernet A-D route the ESI field is set to the CE's ESI and the Ethernet Tag ID is set to the VPWS service instance identifier, which MUST have the same value on all PEs attached to that ES. This allows an ingress PE in a multihoming all-active scenario to perform flow-based load-balancing of traffic flows to all of the PEs attached to that ES. In all cases traffic follows the transport paths, which may be asymmetric.

The VPWS service instance identifier encoded in the Ethernet Tag ID

in an advertised per-EVI Ethernet A-D route MUST either be unique across all ASs, or an ASBR needs to perform a translation when the per-EVI Ethernet A-D route is re-advertised by the ASBR from one AS to the other AS.

A per-ES Ethernet A-D route can be used for mass withdraw to withdraw all per-EVI Ethernet A-D routes associated with the multi-home site on a given PE.

5 EVPN Comparison to PW Signaling

In EVPN, service endpoint discovery and label signaling are done concurrently using BGP. Whereas, with VPWS based on [RFC4448], label signaling is done via LDP and service endpoint discovery is either through manual provisioning or through BGP.

In existing implementations of VPWS using pseudowires(PWs), redundancy is limited to single-active mode, while with EVPN implementation of VPWS both single-active and all-active redundancy modes can be supported.

In existing implementations with PWs, backup PWs are not used to carry traffic, while with EVPN, traffic can be load-balanced among different PEs multi-homed to a single CE.

Upon link or node failure, EVPN can trigger failover with the withdrawal of a single BGP route per EVPL service or multiple EVPL services, whereas with VPWS PW redundancy, the failover sequence requires exchange of two control plane messages: one message to deactivate the group of primary PWs and a second message to activate the group of backup PWs associated with the access link.

Finally, EVPN may employ data plane egress link protection mechanisms not available in VPWS. This can be done by the primary PE (on local AC down) using the label advertised in the per-EVI Ethernet A-D route by the backup PE to encapsulate the traffic and direct it to the backup PE.

6 Failure Scenarios

On a link or port failure between the CE and the PE for both single and multi-homed CEs, unlike [RFC7432] the PE MUST withdraw all the associated Ethernet A-D routes for the VPWS service instances on the failed port or link.

6.1 Single-Homed CEs

Unlike [RFC7432], EVPN-VPWS uses Ethernet A-D route advertisements for single-homed Ethernet Segments. Therefore, upon a link/port failure of this single-homed Ethernet Segment, the PE MUST withdraw the associated per-EVI Ethernet A-D routes.

6.2 Multi-Homed CEs

For a faster convergence in multi-homed scenarios with either Single-Active Redundancy or All-active redundancy, a mass withdraw technique is used. A PE previously advertising a per-ES Ethernet A-D route, can withdraw this route by signaling to the remote PEs to switch all the VPWS service instances associated with this multi-homed ES to the backup PE.

7 Acknowledgements

The authors would like to acknowledge Jeffrey Zhang, Wen Lin, Nitin Singh, Senthil Sathappan, Vinod Prabhu, Himanshu Shah, Iftekhar Hussain, Alvaro Retana and Acee Lindem for their feedback and contributions to this document.

8 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 MPLS and IP-based tunnel technologies to support data plane transport. Security considerations described in [RFC7432] and in [ietf-evpn-overlay] are equally applicable.

9 IANA Considerations

IANA has allocated the following EVPN Extended Community sub-type:

SUB-TYPE VALUE	NAME	Reference
0x04	EVPN Layer 2 Attributes	[RFCXXXX]

This document creates a registry called "EVPN Layer 2 Attributes Control Flags". New registrations will be made through the "RFC Required" procedure defined in [RFC5226].

Initial registrations are as follows:

P	Advertising PE is the Primary PE.
B	Advertising PE is the Backup PE.
C	Control word [RFC4448] MUST be present.

10 References

10.1 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>.

[RFC4448] Martini, L., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, April 2006.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", November 2012.

[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>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, February 2006, <http://www.rfc-editor.org/info/rfc4360>.

[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>.

[RFC7348] Mahalingam, M., et al, "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014

10.2 Informative References

[MEF] Metro Ethernet Forum, "Ethernet Services Definitions - Phase 2", Technical Specification MEF 6.1, April 2008, https://www.mef.net/Assets/Technical_Specifications/PDF/MEF_6.1.pdf

[RFC4664] Andersson, L., Ed., and E. Rosen, Ed., "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, September 2006, <http://www.rfc-editor.org/info/rfc4664>.

[ietf-evpn-overlay] Sajassi-Drake et al., "A Network Virtualization Overlay Solution using EVPN", draft-ietf-bess-evpn-overlay-07.txt, work in progress, December, 2016

Contributors

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

Daniel Voyer Bell Canada

Authors' Addresses

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

Ali Sajassi
Cisco
Email: sajassi@cisco.com

Samer Salam
Cisco
Email: ssalam@cisco.com

John Drake
Juniper Networks
Email: jdrake@juniper.net

Jeff Tantsura
Individual
Email: jefftant@gmail.com

Dirk Steinberg
Steinberg Consulting
Email: dws@steinbergnet.net

Patrice Brissette
Cisco
Email: pbrisset@cisco.com

Thomas Beckhaus
Deutsche Telecom
Email: Thomas.Beckhaus@telekom.de

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

Ryan Bickhart
Juniper Networks

Email: rbickhart@juniper.net

BESS Workgroup
Internet Draft
Intended status: Standards Track

J. Rabadan, Ed.
S. Sathappan
W. Henderickx
Alcatel-Lucent

R. Shekhar
N. Sheth
W. Lin
M. Katiyar
Juniper

A. Sajassi
Cisco

A. Isaac
Bloomberg

M. Tufail
Citibank

Expires: January 7, 2016

July 6, 2015

Optimized Ingress Replication solution for EVPN
draft-rabadan-bess-evpn-optimized-ir-01

Abstract

Network Virtualization Overlay (NVO) networks using EVPN as control plane may use ingress replication (IR) or PIM-based trees to convey the overlay multicast traffic. PIM provides an efficient solution to avoid sending multiple copies of the same packet over the same physical link, however it may not always be deployed in the NVO core network. IR avoids the dependency on PIM in the NVO network core. While IR provides a simple multicast transport, some NVO networks with demanding multicast applications require a more efficient solution without PIM in the core. This document describes a solution to optimize the efficiency of IR in NVO networks.

material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on January 7, 2015.

Table of Contents

1. Problem Statement

EVPN may be used as the control plane for a Network Virtualization Overlay (NVO) network. Network Virtualization Edge (NVE) devices and PEs that are part of the same EVI use Ingress Replication (IR) or PIM-based trees to transport the tenant's multicast traffic. In NVO networks where PIM-based trees cannot be used, IR is the only alternative. Examples of these situations are NVO networks where the core nodes don't support PIM or the network operator does not want to run PIM in the core.

In some use-cases, the amount of replication for BUM (Broadcast, Unknown unicast and Multicast traffic) is kept under control on the NVEs due to the following fairly common assumptions:

a) Broadcast is greatly reduced due to the proxy-ARP and proxy-ND capabilities supported by EVPN on the NVEs. Some NVEs can even provide DHCP-server functions for the attached Tenant Systems (TS) reducing the broadcast even further.

b) Unknown unicast traffic is greatly reduced in virtualized NVO networks where all the MAC and IP addresses are learnt in the control plane.

c) Multicast applications are not used.

If the above assumptions are true for a given NVO network, then IR

provides a simple solution for multi-destination traffic. However, the statement c) above is not always true and multicast applications are required in many use-cases.

When the multicast sources are attached to NVEs residing in hypervisors or low-performance-replication TORs, the ingress replication of a large amount of multicast traffic to a significant number of remote NVEs/PEs can seriously degrade the performance of the NVE and impact the application.

This document describes a solution that makes use of two IR optimizations:

i) Assisted-Replication (AR)
ii) Pruned-Flood-Lists (PFL)

Both optimizations may be used together or independently so that the performance and efficiency of the network to transport multicast can be improved. Both solutions require some extensions to [EVPN] that are described in section 3.

Section 2 lists the requirements of the combined optimized-IR solution, whereas sections 4 and 5 describe the Assisted-Replication (AR) solution, and section 6 the Pruned-Flood-Lists (PFL) solution.

2. Solution requirements

The IR optimization solution (optimized-IR hereafter) MUST meet the following requirements:

a) The solution MUST provide an IR optimization for BM (Broadcast and Multicast) traffic, while preserving the packet order for unicast applications, i.e. known and unknown unicast traffic SHALL follow the same path.

b) The solution MUST be compatible with [EVPN] and [EVPN-OVERLAY] and not have any impact on the EVPN procedures for BM traffic. In particular, the solution MUST support the following EVPN functions:

   - All-active multi-homing, including the split-horizon and Designated Forwarder (DF) functions.

   - Single-active multi-homing, including the DF function.

   - Handling of multi-destination traffic and processing of broadcast and multicast as per [EVPN].

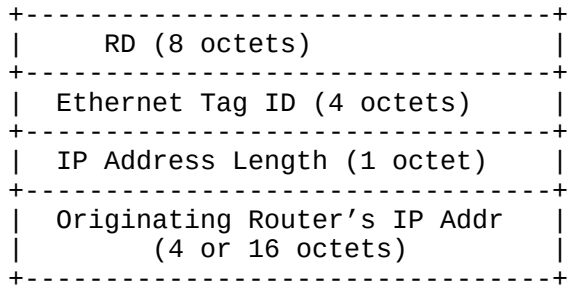
c) The solution MUST be backwards compatible with existing NVEs using a non-optimized version of IR. A given EVI can have NVEs/PEs supporting regular-IR and optimized-IR.

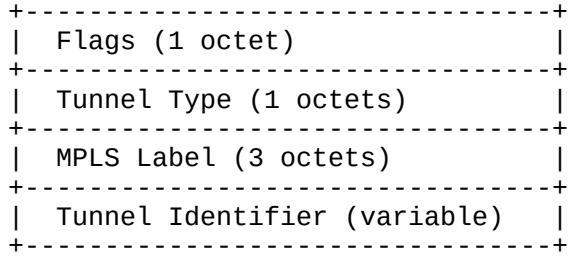
d) The solution MUST be independent of the NVO specific data plane encapsulation and the virtual identifiers being used, e.g.: VXLAN VNIs, NVGRE VSIDs or MPLS labels.

3. EVPN BGP Attributes for optimized-IR

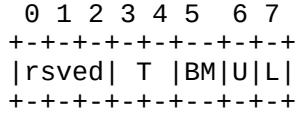
This solution proposes some changes to the [EVPN] Inclusive Multicast Ethernet Tag routes and attributes so that an NVE/PE can signal its optimized-IR capabilities.

The Inclusive Multicast Ethernet Tag route (RT-3) and its PMSI Tunnel Attribute's (PTA) general format used in [EVPN] are shown below:





The Flags field is defined as follows:



Where a new type field (for AR) and two new flags (for PFL signaling) are defined:

- T is the AR Type field (2 bits) that defines the AR role of the advertising router:

  + 00 (decimal 0) = RNVE (non-AR support)

  + 01 (decimal 1) = AR-REPLICATOR

  + 10 (decimal 2) = AR-LEAF

- The PFL (Pruned-Flood-Lists) flags defined the desired behavior of the advertising router for the different types of traffic:

  + BM= Broadcast and Multicast (BM) flag. BM=1 means "prune-me" from the BM flooding list. BM=0 means regular behavior.

  + U= Unknown flag. U=1 means "prune-me" from the Unknown flooding list. U=0 means regular behavior.

- Flag L is an existing flag defined in [RFC6514] (L=Leaf Information Required) and it will be used only in the Selective AR Solution.

Please refer to section 10 for the IANA considerations related to the PTA flags.

In this document, the above RT-3 and PTA can be used in three different modes for the same EVI/Ethernet Tag:

o Regular-IR route: in this route, Originating Router’s IP Address, Tunnel Type (0x06), MPLS Label, Tunnel Identifier and Flags MUST be used as described in [EVPN]. The Originating Router’s IP Address and Tunnel Identifier are set to an IP address that we denominate IR-IP in this document.

o Replicator-AR route: this route is used by the AR-REPLICATOR to advertise its AR capabilities, with the fields set as follows.

  + Originating Router’s IP Address as well as the Tunnel Identifier are set to the same routable IP address that we denominate AR-IP and SHOULD be different than the IR-IP for a given PE/NVE.

  + Tunnel Type = Assisted-Replication (AR). Section 11 provides the allocated type value.

  + T (AR role type) = 01 (AR-REPLICATOR).

  + L (Leaf Information Required) = 0 (for non-selective AR) or 1 (for selective AR).

- o Leaf-AR route: this route MAY be used by the AR-LEAF to advertise its desire to receive the multicast traffic from a specific AR-REPLICATOR. It is only used for selective AR and its fields are set as follows:
  - + Originating Router’s IP Address is set to the advertising IR-IP (same IP used by the AR-LEAF in regular-IR routes).
  - + Tunnel Identifier is set to the AR-IP of the AR-REPLICATOR from which the multicast traffic is requested.
  - + Tunnel Type = Assisted-Replication (AR). Section 11 provides the allocated type value.
  - + T (AR role type) = 02 (AR-LEAF).

Each AR-enabled node MUST understand and process the AR type field in the PTA (Flags field) of replicator-AR and leaf-AR routes, and MUST signal the corresponding type (1 or 2) according to its administrative choice for replicator-AR and leaf-AR routes.

Each node, part of the EVI, MAY understand and process the BM/U flags. Note that these BM/U flags may be used to optimize the delivery of multi-destination traffic and its use SHOULD be an administrative choice, and independent of the AR role.

Non-optimized-IR nodes will be unaware of the new PMSI attribute flag definition as well as the new Tunnel Type (AR), i.e. they will ignore the information contained in the flags field for any RT-3 and will ignore the RT-3 routes with an unknown Tunnel Type (type AR in this case).

4. Non-selective Assisted-Replication (AR) Solution Description

The following figure illustrates an example NVO network where the non-selective AR function is enabled. Three different roles are defined for a given EVI: AR-REPLICATOR, AR-LEAF and RNVE (Regular NVE). The solution is called "non-selective" because the chosen AR-REPLICATOR for a given flow MUST replicate the multicast traffic to ’all’ the NVE/PEs in the EVI except for the source NVE/PE.

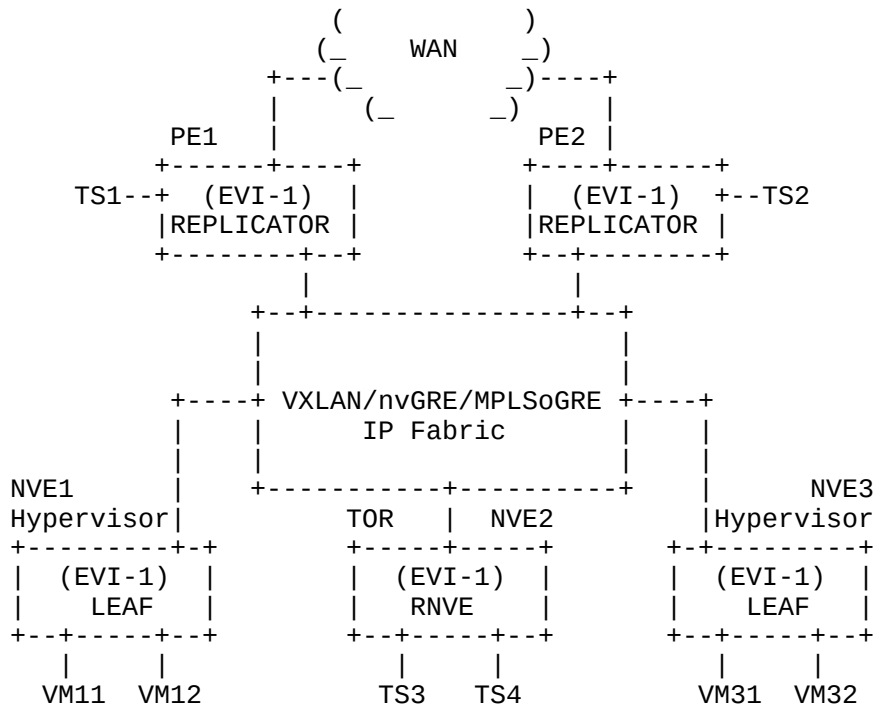


Figure 1 Optimized-IR scenario

4.1. Non-selective AR-REPLICATOR procedures

An AR-REPLICATOR is defined as an NVE/PE capable of replicating ingress BM (Broadcast and Multicast) traffic received on an overlay tunnel to other overlay tunnels and local Attachment Circuits (ACs). The AR-REPLICATOR signals its role in the control plane and understands where the other roles (AR-LEAF nodes, RNVEs and other AR-REPLICATORs) are located. A given AR-enabled EVI service may have zero, one or more AR-REPLICATORs. In our example in figure 1, PE1 and PE2 are defined as AR-REPLICATORs. The following considerations apply to the AR-REPLICATOR role:

a) The AR-REPLICATOR role SHOULD be an administrative choice in any NVE/PE that is part of an AR-enabled EVI. This administrative option to enable AR-REPLICATOR capabilities MAY be implemented as a system level option as opposed to as a per-EVI option.

b) An AR-REPLICATOR MUST advertise a Replicator-AR route and MAY advertise a Regular-IR route. The AR-REPLICATOR MUST NOT generate a Regular-IR route if it does not have local attachment circuits

(AC).

c) The Replicator-AR and Regular-IR routes will be generated according to section 3. The AR-IP and IR-IP used by the Replicator-AR will be different routable IP addresses.

d) When a node defined as AR-REPLICATOR receives a packet on an overlay tunnel, it will do a tunnel destination IP lookup and apply the following procedures:

- If the destination IP is the AR-REPLICATOR IR-IP Address the node will process the packet normally as in [EVPN].

- If the destination IP is the AR-REPLICATOR AR-IP Address the node MUST replicate the packet to local ACs and overlay tunnels (excluding the overlay tunnel to the source of the packet). When replicating to remote AR-REPLICATORs the tunnel destination IP will be an IR-IP. That will be an indication for the remote AR-REPLICATOR that it MUST NOT replicate to overlay tunnels. The tunnel source IP will be the AR-IP of the AR-REPLICATOR.

4.2. Non-selective AR-LEAF procedures

AR-LEAF is defined as an NVE/PE that - given its poor replication performance - sends all the BM traffic to an AR-REPLICATOR that can replicate the traffic further on its behalf. It MAY signal its AR-LEAF capability in the control plane and understands where the other roles are located (AR-REPLICATOR and RNVEs). A given service can have zero, one or more AR-LEAF nodes. Figure 1 shows NVE1 and NVE2 (both residing in hypervisors) acting as AR-LEAF. The following considerations apply to the AR-LEAF role:

a) The AR-LEAF role SHOULD be an administrative choice in any NVE/PE that is part of an AR-enabled EVI. This administrative option to enable AR-LEAF capabilities MAY be implemented as a system level option as opposed to as per-EVI option.

b) In this non-selective AR solution, the AR-LEAF MUST advertise a single Regular-IR inclusive multicast route as in [EVPN].

c) In a service where there are no AR-REPLICATORs, the AR-LEAF MUST use regular ingress replication. This will happen when a new update from the last former AR-REPLICATOR is received and contains a non-REPLICATOR AR type, or when the AR-LEAF detects that the last AR-REPLICATOR is down (next-hop tracking in the IGP or any other detection mechanism). Ingress replication MUST use the

forwarding information given by the remote Regular-IR Inclusive Multicast Routes as described in [EVPN].

d) In a service where there is one or more AR-REPLICATORs (based on the received Replicator-AR routes for the EVI), the AR-LEAF can locally select which AR-REPLICATOR it sends the BM traffic to:

- o A single AR-REPLICATOR MAY be selected for all the BM packets received on the AR-LEAF attachment circuits (ACs) for a given EVI. This selection is a local decision and it does not have to match other AR-LEAF's selection within the same EVI.
- o An AR-LEAF MAY select more than one AR-REPLICATOR and do either per-flow or per-EVI load balancing.
- o In case of a failure on the selected AR-REPLICATOR, another AR-REPLICATOR will be selected.
- o When an AR-REPLICATOR is selected, the AR-LEAF MUST send all the BM packets to that AR-REPLICATOR using the forwarding information given by the Replicator-AR route for the chosen AR-REPLICATOR, with tunnel type = TBD (AR tunnel). The underlay destination IP address MUST be the AR-IP advertised by the AR-REPLICATOR in the Replicator-AR route.
- o AR-LEAF nodes SHALL send service-level BM control plane packets following regular IR procedures. An example would be IGMP, MLD or PIM multicast packets. The AR-REPLICATORs MUST not replicate these control plane packets to other overlay tunnels since they will use the regular IR-IP Address.

4.3. RNVE procedures

RNVE (Regular Network Virtualization Edge node) is defined as an NVE/PE without AR-REPLICATOR or AR-LEAF capabilities that does IR as described in [EVPN]. The RNVE does not signal any AR role and is unaware of the AR-REPLICATOR/LEAF roles in the EVI. The RNVE will ignore the Flags in the Regular-IR routes and will ignore the Replicator-AR and Leaf-AR routes entirely (due to an unknown tunnel type in the PTA).

This role provides EVPN with the backwards compatibility required in optimized-IR EVIs. Figure 1 shows NVE2 as RNVE.

4.4. Forwarding behavior in non-selective AR EVIs

In AR EVIs, BM (Broadcast and Multicast) traffic between two NVEs may

follow a different path than unicast traffic. This solution proposes the replication of BM through the AR-REPLICATOR node, whereas unknown/known unicast will be delivered directly from the source node to the destination node without being replicated by any intermediate node. Unknown unicast SHALL follow the same path as known unicast traffic in order to avoid packet reordering for unicast applications and simplify the control and data plane procedures. Section 4.4.1. describes the expected forwarding behavior for BM traffic in nodes acting as AR-REPLICATOR, AR-LEAF and RNVE. Section 4.4.2. describes the forwarding behavior for unknown unicast traffic.

Note that known unicast forwarding is not impacted by this solution.

4.4.1. Broadcast and Multicast forwarding behavior

The expected behavior per role is described in this section.

4.4.1.1. Non-selective AR-REPLICATOR BM forwarding

The AR-REPLICATORs will build a flooding list composed of ACs and overlay tunnels to remote nodes in the EVI. Some of those overlay tunnels MAY be flagged as non-BM receivers based on the BM flag received from the remote nodes in the EVI.

- When an AR-REPLICATOR receives a BM packet on an AC, it will forward the BM packet to its flooding list (including local ACs and remote NVE/PEs), skipping the non-BM overlay tunnels.

- When an AR-REPLICATOR receives a BM packet on an overlay tunnel, it will check the destination IP of the underlay IP header and:

  - If the destination IP matches its AR-IP, the AR-REPLICATOR will forward the BM packet to its flooding list (ACs and overlay tunnels) excluding the non-BM overlay tunnels. The AR-REPLICATOR will do source squelching to ensure the traffic is not sent back to the originating AR-LEAF. If the overlay encapsulation is MPLS and the EVI label is not the bottom of the stack, the AR-REPLICATOR MUST copy the rest of the labels and forward them to the egress overlay tunnels.

  - If the destination IP matches its IR-IP, the AR-REPLICATOR will skip all the overlay tunnels from the flooding list, i.e. it will only replicate to local ACs. This is the regular IR behavior described in [EVPN].

4.4.1.2. Non-selective AR-LEAF BM forwarding

The AR-LEAF nodes will build two flood-lists:

1) Flood-list #1 - composed of ACs and an AR-REPLICATOR-set of overlay tunnels. The AR-REPLICATOR-set is defined as one or more overlay tunnels to the AR-IP Addresses of the remote AR-REPLICATOR(s) in the EVI. The selection of more than one AR-REPLICATOR is described in section 4.2. and it is a local AR-LEAF decision.

2) Flood-list #2 - composed of ACs and overlay tunnels to the remote IR-IP Addresses.

When an AR-LEAF receives a BM packet on an AC, it will check the AR-REPLICATOR-set:

- If the AR-REPLICATOR-set is empty, the AR-LEAF will send the packet to flood-list #2.

- If the AR-REPLICATOR-set is NOT empty, the AR-LEAF will send the packet to flood-list #1, where only one of the overlay tunnels of the AR-REPLICATOR-set is used.

When an AR-LEAF receives a BM packet on an overlay tunnel, will forward the BM packet to its local ACs and never to an overlay tunnel. This is the regular IR behavior described in [EVPN].

4.4.1.3. RNVE BM forwarding

The RNVE is completely unaware of the AR-REPLICATORs, AR-LEAF nodes and BM/U flags (that information is ignored). Its forwarding behavior is the regular IR behavior described in [EVPN]. Any regular non-AR node is fully compatible with the RNVE role described in this document.

4.4.2. Unknown unicast forwarding behavior

The expected behavior is described in this section.

4.4.2.1. Non-selective AR-REPLICATOR/LEAF Unknown unicast forwarding

While the forwarding behavior in AR-REPLICATORs and AR-LEAF nodes is different for BM traffic, as far as Unknown unicast traffic forwarding is concerned, AR-LEAF nodes behave exactly in the same way as AR-REPLICATORs do.

The AR-REPLICATOR/LEAF nodes will build a flood-list composed of ACs and overlay tunnels to the IR-IP Addresses of the remote nodes in the EVI. Some of those overlay tunnels MAY be flagged as non-U (Unknown

unicast) receivers based on the U flag received from the remote nodes in the EVI.

- o When an AR-REPLICATOR/LEAF receives an unknown packet on an AC, it will forward the unknown packet to its flood-list, skipping the non-U overlay tunnels.

- o When an AR-REPLICATOR/LEAF receives an unknown packet on an overlay tunnel will forward the unknown packet to its local ACs and never to an overlay tunnel. This is the regular IR behavior described in [EVPN].

4.4.2.2. RNVE Unknown unicast forwarding

As described for BM traffic, the RNVE is completely unaware of the REPLICATORs, LEAF nodes and BM/U flags (that information is ignored). Its forwarding behavior is the regular IR behavior described in [EVPN], also for Unknown unicast traffic. Any regular non-AR node is fully compatible with the RNVE role described in this document.

5. Selective Assisted-Replication (AR) Solution Description

Figure 1 is also used to describe the selective AR solution, however in this section we consider NVE2 as one more AR-LEAF for EVI-1. The solution is called "selective" because a given AR-REPLICATOR MUST replicate the BM traffic to only the AR-LEAF that requested the replication (as opposed to all the AR-LEAF nodes) and MAY replicate the BM traffic to the RNVEs. The same AR roles defined in section 4 are used here, however the procedures are slightly different.

The following sub-sections describe the differences in the procedures of AR-REPLICATOR/LEAFs compared to the non-selective AR solution. There is no change on the RNVEs.

5.1. Selective AR-REPLICATOR procedures

In our example in figure 1, PE1 and PE2 are defined as Selective AR-REPLICATORs. The following considerations apply to the Selective AR-REPLICATOR role:

a) The Selective AR-REPLICATOR capability SHOULD be an administrative choice in any NVE/PE that is part of an AR-enabled EVI, as the AR role itself. This administrative option MAY be implemented as a system level option as opposed to as a per-EVI option.

b) Each AR-REPLICATOR will build a list of AR-REPLICATOR, AR-LEAF and

RNVE nodes (AR-LEAF nodes that sent only a regular-IR route are accounted as RNVEs by the AR-REPLICATOR). In spite of the 'Selective' administrative option, an AR-REPLICATOR MUST NOT behave as a Selective AR-REPLICATOR if at least one of the AR-REPLICATORs has the L flag NOT set. If at least one AR-REPLICATOR sends a Replicator-AR route with L=0 (in the EVI context), the rest of the AR-REPLICATORs will fall back to non-selective AR mode.

b) The Selective AR-REPLICATOR MUST follow the procedures described in section 4.1, except for the following differences:

- The Replicator-AR route MUST include L=1 (Leaf Information Required) in the Replicator-AR route. This flag is used by the AR-REPLICATORs to advertise their 'selective' AR-REPLICATOR capabilities.

- The AR-REPLICATOR will build a 'selective' AR-LEAF-set with the list of nodes that requested replication to its own AR-IP. For instance, assuming NVE1 and NVE2 advertise a Leaf-AR route with PE1's AR-IP (as Tunnel Identifier) and NVE3 advertises a Leaf-AR route with PE2's AR-IP, PE1 MUST only add NVE1/NVE2 in its selective AR-LEAF-set for EVI-1, and exclude NVE3.

- When a node defined and operating as Selective AR-REPLICATOR receives a packet on an overlay tunnel, it will do a tunnel destination IP lookup and if the destination IP is the AR-REPLICATOR AR-IP Address, the node MUST replicate the packet to:

  + local ACs
  + overlay tunnels in the Selective AR-LEAF-set (excluding the overlay tunnel to the source AR-LEAF).
  + overlay tunnels to the RNVEs if the tunnel source IP is the IR-IP of an AR-LEAF (in any other case, the AR-REPLICATOR MUST NOT replicate the BM traffic to remote RNVEs). In other words, the first-hop selective AR-REPLICATOR will replicate to all the RNVEs.
  + overlay tunnels to the remote Selective AR-REPLICATORs if the tunnel source IP is the IR-IP of its own AR-LEAF-set (in any other case, the AR-REPLICATOR MUST NOT replicate the BM traffic to remote AR-REPLICATORs), where the tunnel destination IP is the AR-IP of the remote Selective AR-REPLICATOR. The tunnel destination IP AR-IP will be an indication for the remote Selective AR-REPLICATOR that the packet needs further replication to its AR-LEAFs.

5.2. Selective AR-LEAF procedures

A Selective AR-LEAF chooses a single Selective AR-REPLICATOR per EVI and:

- Sends all the EVI BM traffic to that AR-REPLICATOR and
- Expects to receive the BM traffic for a given EVI from the same AR-REPLICATOR.

In the example of Figure 1, we consider that NVE1/NVE2/NVE3 as Selective AR-LEAFs. NVE1 selects PE1 as its Selective AR-REPLICATOR. If that is so, NVE1 will send all its BM traffic for EVI-1 to PE1. If other AR-LEAF/REPLICATORs send BM traffic, NVE1 will receive that traffic from PE1. These are the differences in the behavior of a Selective AR-LEAF compared to a non-selective AR-LEAF:

a) The AR-LEAF role selective capability SHOULD be an administrative choice in any NVE/PE that is part of an AR-enabled EVI. This administrative option to enable AR-LEAF capabilities MAY be implemented as a system level option as opposed to as per-EVI option.

b) The AR-LEAF MAY advertise a Regular-IR route if there are RNVEs or non-selective AR-LEAFs in the EVI. The Selective AR-LEAF MUST advertise a Leaf-AR route after receiving a Replicator-AR route with L=1. It is recommended that the Selective AR-LEAF waits for a timer t before sending the Leaf-AR route, so that the AR-LEAF receives all the Replicator-AR routes for the EVI.

c) In a service where there is more than one Selective AR-REPLICATORs the Selective AR-LEAF MUST locally select a single Selective AR-REPLICATOR for the EVI. Once selected:

   - The Selective AR-LEAF will send a Leaf-AR route including the AR-IP of the selected AR-REPLICATOR.

   - The Selective AR-LEAF will send all the BM packets received on the attachment circuits (ACs) for a given EVI to that AR-REPLICATOR.

   - In case of a failure on the selected AR-REPLICATOR, another AR-REPLICATOR will be selected and a new Leaf-AR update will be issued, including the new AR-IP. This new route will update the selective list in the new Selective AR-REPLICATOR. In case of failure on the active Selective AR-REPLICATOR, it is recommended for the Selective AR-LEAF to revert to IR behavior for a timer t to speed up the convergence. When the timer expires, the Selective AR-LEAF will resume its AR mode with

the new Selective AR-REPLICATOR.

5.3. Forwarding behavior in selective AR EVIs

This section describes the differences of the selective AR forwarding mode compared to the non-selective mode. Compared to section 4.4, there are no changes for the forwarding behavior in RNVEs or for unknown unicast traffic.

5.3.1. Selective AR-REPLICATOR BM forwarding

The Selective AR-REPLICATORs will build two flood-lists:

1) Flood-list #1 - composed of ACs and overlay tunnels to the remote nodes in the EVI, always using the IR-IPs in the tunnel destination IP addresses. Some of those overlay tunnels MAY be flagged as non-BM receivers based on the BM flag received from the remote nodes in the EVI.

2) Flood-list #2 - composed of ACs, a Selective AR-LEAF-set and a Selective AR-REPLICATOR-set, where:

   o The Selective AR-LEAF-set is composed of the overlay tunnels to the AR-LEAFs that advertise a Leaf-AR route with the AR-IP of the local AR-REPLICATOR. This set is updated with every Leaf-AR route received with a change in the AR-IP included in the PTA’s Tunnel Identifier.

   o The Selective AR-REPLICATOR-set is composed of the overlay tunnels to all the AR-REPLICATORs that send a Replicator-AR route with L=1. The AR-IP addresses are used as tunnel destination IP.

When a Selective AR-REPLICATOR receives a BM packet on an AC, it will forward the BM packet to its flood-list #1, skipping the non-BM overlay tunnels.

When a Selective AR-REPLICATOR receives a BM packet on an overlay tunnel, it will check the destination and source IPs of the underlay IP header and:

- If the destination IP matches its AR-IP and the source IP matches an IP of its own Selective AR-LEAF-set, the AR-REPLICATOR will forward the BM packet to its flood-list #2, as long as the list of AR-REPLICATORs for the EVI matches the Selective AR-REPLICATOR-set. If the Selective AR-REPLICATOR-set does not match the list of AR-REPLICATORs, the node reverts back

to non-selective mode and flood-list #1 is used.

- If the destination IP matches its AR-IP and the source IP does not match any IP of its Selective AR-LEAF-set, the AR-REPLICATOR will forward the BM packet to flood-list #2 but skipping the AR-REPLICATOR-set.

- If the destination IP matches its IR-IP, the AR-REPLICATOR will use flood-list #1 but MUST skip all the overlay tunnels from the flooding list, i.e. it will only replicate to local ACs. This is the regular-IR behavior described in [EVPN].

In any case, non-BM overlay tunnels are excluded from flood-lists and also source squelching is always done in order to ensure the traffic is not sent back to the originating source. If the overlay encapsulation is MPLS and the EVI label is not the bottom of the stack, the AR-REPLICATOR MUST copy the rest of the labels when forwarding them to the egress overlay tunnels.

5.3.2. Selective AR-LEAF BM forwarding

The Selective AR-LEAF nodes will build two flood-lists:

1) Flood-list #1 - composed of ACs and the overlay tunnel to the selected AR-REPLICATOR (using the AR-IP as the tunnel destination IP).

2) Flood-list #2 - composed of ACs and overlay tunnels to the remote IR-IP Addresses.

When an AR-LEAF receives a BM packet on an AC, it will check if there is any selected AR-REPLICATOR. If there is, flood-list #1 will be used. Otherwise, flood-list #2 will.

When an AR-LEAF receives a BM packet on an overlay tunnel, will forward the BM packet to its local ACs and never to an overlay tunnel. This is the regular IR behavior described in [EVPN].

6. Pruned-Flood-Lists (PFL)

In addition to AR, the second optimization supported by this solution is the ability for the all the EVI nodes to signal Pruned-Flood-Lists (PFL). As described in section 3, an EVPN node can signal a given value for the BM and U PFL flags in the IR Inclusive Multicast Routes, where:

+ BM= Broadcast and Multicast (BM) flag. BM=1 means "prune-me" from the BM flood-list. BM=0 means regular behavior.

+ U= Unknown flag. U=1 means "prune-me" from the Unknown flood-list. U=0 means regular behavior.

The ability to signal these PFL flags is an administrative choice. Upon receiving a non-zero PFL flag, a node MAY decide to honor the PFL flag and remove the sender from the corresponding flood-list. A given EVI node receiving BUM traffic on an overlay tunnel MUST replicate the traffic normally, regardless of the signaled PFL flags.

This optimization MAY be used along with the AR solution.

6.1. A PFL example

In order to illustrate the use of the solution described in this document, we will assume that EVI-1 in figure 1 is optimized-IR enabled and:

- o PE1 and PE2 are administratively configured as AR-REPLICATORs, due to their high-performance replication capabilities. PE1 and PE2 will send a Replicator-AR route with BM/U flags = 00.

- o NVE1 and NVE3 are administratively configured as AR-LEAF nodes, due to their low-performance software-based replication capabilities. They will advertise a Leaf-AR route. Assuming both NVEs advertise all the attached VMs in EVPN as soon as they come up and don’t have any VMs interested in multicast applications, they will be configured to signal BM/U flags = 11 for EVI-1.

- o NVE2 is optimized-IR unaware; therefore it takes on the RNVE role in EVI-1.

Based on the above assumptions the following forwarding behavior will take place:

(1) Any BM packets sent from VM11 will be sent to VM12 and PE1. PE1 will forward further the BM packets to TS1, WAN link, PE2 and NVE2, but not to NVE3. PE2 and NVE2 will replicate the BM packets to their local ACs but we will avoid NVE3 having to replicate unnecessarily those BM packets to VM31 and VM32.

(2) Any BM packets received on PE2 from the WAN will be sent to PE1 and NVE2, but not to NVE1 and NVE3, sparing the two hypervisors from replicating unnecessarily to their local VMs. PE1 and NVE2 will replicate to their local ACs only.

(3) Any Unknown unicast packet sent from VM31 will be forwarded by NVE3 to NVE2, PE1 and PE2 but not NVE1. The solution avoids the unnecessary replication to NVE1, since the destination of the unknown traffic cannot be at NVE1.

(4) Any Unknown unicast packet sent from TS1 will be forwarded by PE1 to the WAN link, PE2 and NVE2 but not to NVE1 and NVE3, since the target of the unknown traffic cannot be at those NVEs.

7. AR Procedures for single-IP AR-REPLICATORS

The procedures explained in sections 4 (Non-selective AR) and 5 (Selective AR) assume that the AR-REPLICATOR can use two local routable IP addresses to terminate and initiate NVO tunnels, i.e. IR-IP and AR-IP addresses. This is usually the case for PE-based AR-REPLICATOR nodes.

In some cases, the AR-REPLICATOR node does not support more than one IP address to terminate and initiate NVO tunnels, i.e. the IR-IP and AR-IP are the same IP addresses. This may be the case in some software-based or low-end AR-REPLICATOR nodes. If this is the case, the procedures in sections 4 and 5 must be modified in the following way:

- The Replicator-AR routes generated by the AR-REPLICATOR use an AR-IP that will match its IR-IP. In order to differentiate the data plane packets that need to use IR from the packets that must use AR forwarding mode, the Replicator-AR route must advertise a different VNI/VSID than the one used by the Regular-IR route. For instance, the AR-REPLICATOR will advertise AR-VNI along with the Replicator-AR route and IR-VNI along with the Regular-IR route. Since both routes have the same key, different RDs are needed for both routes.

- An AR-REPLICATOR will perform IR or AR forwarding mode for the incoming Overlay packets based on an ingress VNI lookup, as opposed to the tunnel IP DA lookup described in sections 4 and 5. Note that, when replicating to remote AR-REPLICATOR nodes, the use of the IR-VNI or AR-VNI advertised by the egress node will determine the IR or AR forwarding mode at the subsequent AR-REPLICATOR.

The rest of the procedures will follow what is described in sections 4 and 5.

8. AR Procedures and EVPN Multi-homing Split-Horizon

When EVPN is used for MPLS over GRE, all the multi-homing procedures are compatible with sections 4 and 5 of this document.

If VXLAN or NVGRE are used, and if the Split-horizon is based on the tunnel IP SA and "Local-Bias" as described in [EVPN-OVERLAY], the Split-horizon check will not work if there is an Ethernet-Segment shared between two AR-LEAF nodes, and the AR-REPLICATOR changes the tunnel IP SA of the packets with its own AR-IP.

In order to be compatible with the IP SA split-horizon check, the AR-REPLICATOR MAY keep the original received tunnel IP SA when replicating packets to a remote AR-LEAF or AR-REPLICATOR. This will allow DF (Designated Forwarder) AR-LEAF nodes to apply Split-horizon check procedures for BM packets, before sending them to the local Ethernet-Segment.

Note that if the AR-REPLICATOR implementation keeps the received tunnel IP SA, the use of uRPF in the IP fabric based on the tunnel IP SA MUST be disabled.

9. Out-of-band distribution of Broadcast/Multicast traffic

The use of out-of-band mechanisms to distribute BM traffic between AR-REPLICATORS MAY be used. Details will be provided in future versions of this document.

10. Benefits of the optimized-IR solution

A solution for the optimization of Ingress Replication in EVPN is described in this document (optimized-IR). The solution brings the following benefits:

- Optimizes the multicast forwarding in low-performance NVEs, by relaying the replication to high-performance NVEs (AR-REPLICATORs) and while preserving the packet ordering for unicast applications.
- Reduces the flooded traffic in NVO networks where some NVEs do not need broadcast/multicast and/or unknown unicast traffic.
- It is fully compatible with existing EVPN implementations and EVPN functions for NVO overlay tunnels. Optimized-IR NVEs and regular NVEs can be even part of the same EVI.
- It does not require any PIM-based tree in the NVO core of the network.

11. Conventions used in this document

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

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

In this document, the characters ">>" preceding an indented line(s) indicates a compliance requirement statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the explicit compliance requirements of this RFC.

12. Security Considerations

This section will be added in future versions.

13. IANA Considerations

A new Tunnel-Type (AR) must be requested and allocated by IANA for the PTA (PMSI Tunnel Attribute) used in this document.

In addition to the new Tunnel-Type, this document requests the allocation of the PTA flags as in section 3. A registry is created as per [PTA-FLAGS].

14. Terminology

Regular-IR: Refers to Regular Ingress Replication, where the source NVE/PE sends a copy to each remote NVE/PE part of the EVI.

AR-IP: IP address owned by the AR-REPLICATOR and used to differentiate the ingress traffic that must follow the AR procedures.

IR-IP: IP address used for Ingress Replication as in [EVPN].

AR-VNI: VNI advertised by the AR-REPLICATOR along with the Replicator-AR route. It is used to identify the ingress packets that must follow AR procedures ONLY in the Single-IP AR-REPLICATOR case.

IR-VNI: VNI advertised along with the RT-3 for IR.

AR forwarding mode: for an AR-LEF, it means sending an AC BM packet to a single AR-REPLICATOR with tunnel destination IP AR-IP. For an AR-REPLICATOR, it means sending a BM packet to a selective number or all the overlay tunnels when the packet

was previously received from an overlay tunnel.

IR forwarding mode: it refers to the Ingress Replication behavior explained in [EVPN]. It means sending an AC BM packet copy to each remote PE/NVE in the EVI and sending an overlay BM packet only to the ACs and not other overlay tunnels.

PTA: PMSI Tunnel Attribute

RT-3: EVPN Route Type 3, Inclusive Multicast Ethernet Tag route

15. References

15.1 Normative References

[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, <http://www.rfc-editor.org/info/rfc6514>.

[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>.

15.2 Informative References

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

[PTA-FLAGS] Rosen, E., "IANA Registry for P-Multicast Service Interface Tunnel Attribute Flags", draft-ietf-bess-pta-flags-00.txt, work in progress, February 2015

16. Acknowledgments

The authors would like to thank Neil Hart, David Motz, Thomas Morin and Jeffrey Zhang for their valuable feedback and contributions.

17. Authors’ Addresses

Jorge Rabadan (Editor)
Alcatel-Lucent
777 E. Middlefield Road
Mountain View, CA 94043 USA
Email: jorge.rabadan@alcatel-lucent.com

Senthil Sathappan
Alcatel-Lucent
Email: senthil.sathappan@alcatel-lucent.com

Mukul Katiyar
Juniper
Email: mkatiyar@juniper.net

Wim Henderickx
Alcatel-Lucent
Email: wim.henderickx@alcatel-lucent.com

Ravi Shekhar
Juniper Networks
Email: rshekhar@juniper.net

Nischal Sheth
Juniper Networks
Email: nsheth@juniper.net

Wen Lin
Juniper Networks
Email: wlin@juniper.net

Ali Sajassi
Cisco
Email: sajassi@cisco.com

Aldrin Isaac
Bloomberg
Email: aisaac71@bloomberg.net

Mudassir Tufail
Citibank
mudassir.tufail@citi.com

Internet Working Group
Internet Draft
Category: Standards Track

A. Sajassi
P. Brissette
Cisco
R. Schell
Verizon
J. Drake
Juniper
J. Rabadan
Nokia

Expires: August 26, 2016 February 26, 2018

EVPN Virtual Ethernet Segment
draft-sajassi-bess-evpn-virtual-eth-segment-03

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

Abstract

EVPN and PBB-EVPN introduce a family of solutions for multipoint Ethernet services over MPLS/IP network with many advanced capabilities among which their multi-homing capabilities. These solutions define two types of multi-homing for an Ethernet Segment (ES): 1) Single-Active and 2) All-Active, where an Ethernet Segment is defined as a set of links between the multi-homed device/network and the set of PE devices that they are connected to.

Some Service Providers want to extend the concept of the physical links in an ES to Ethernet Virtual Circuits (EVCs) where many of such EVCs can be aggregated on a single physical External Network-to-Network Interface (ENNI). An ES that consists of a set of EVCs instead of physical links is referred to as a virtual ES (vES). This draft describes the requirements and the extensions needed to support vES in EVPN and PBB-EVPN.

Conventions

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.

Table of Contents

1. Introduction

[EVPN] and [PBB-EVPN] introduce a family of solutions for multipoint Ethernet services over MPLS/IP network with many advanced capabilities among which their multi-homing capabilities. These solutions define two types of multi-homing for an Ethernet Segment (ES): 1) Single-Active and 2) All-Active, where an Ethernet Segment is defined as a set of links between the multi-homed device/network and the set of PE devices that they are connected to.

This document extends the Ethernet Segment concept so that an ES can be associated to a set of EVCs or other objects such as MPLS Label Switch Paths (LSP) or Pseudowires (PW).

1.1 Virtual Ethernet Segments in Access Ethernet Networks

Some Service Providers (SPs) want to extend the concept of the physical links in an ES to Ethernet Virtual Circuits (EVCs) where many of such EVCs can be aggregated on a single physical External Network-to-Network Interface (ENNI). An ES that consists of a set of EVCs instead of physical links is referred to as a virtual ES (vES). Figure below depicts two PE devices (PE1 and PE2) each with an ENNI where a number of vES’s are aggregated on - each of which through its associated EVC.

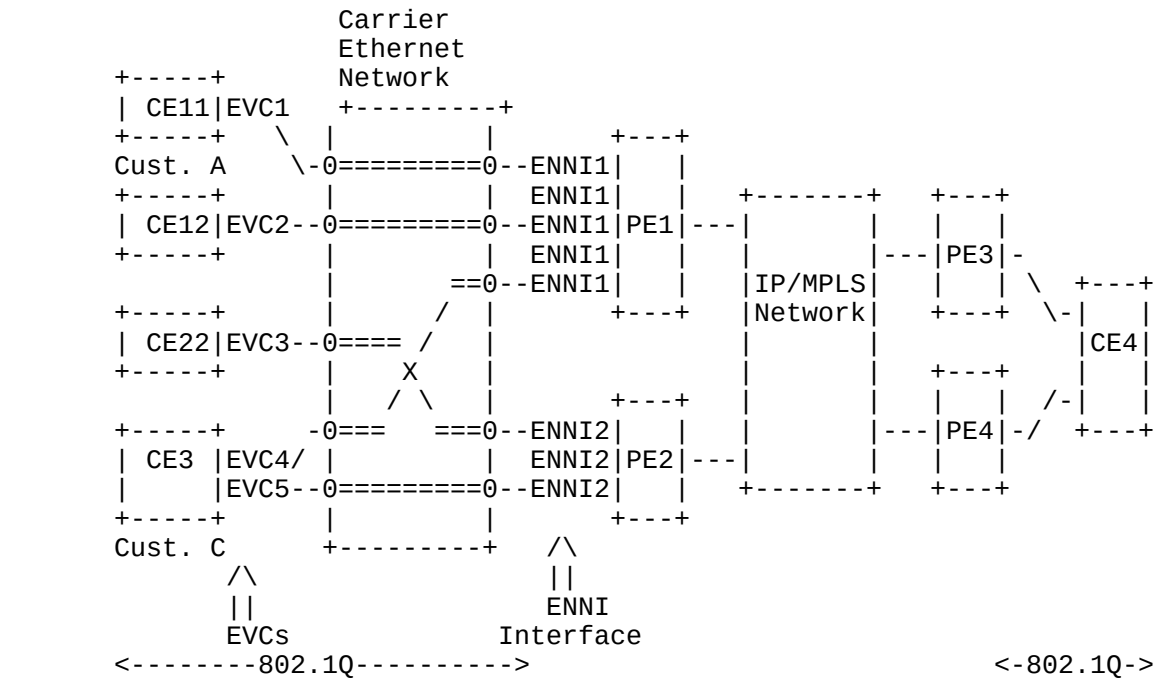


Figure 1: DHD/DHN (both SA/AA) and SH on same ENNI

E-NNIs are commonly used to reach off-network / out-of-franchise customer sites via independent Ethernet access networks or third-party Ethernet Access Providers (EAP) (see above figure). E-NNIs can aggregate traffic from hundreds to thousands of vES's; where, each vES is represented by its associated EVC on that ENNI. As a result, ENNIs and their associated EVCs are a key element of SP off-networks that are carefully designed and closely monitored.

In order to meet customer's Service Level Agreements (SLA), SPs build redundancy via multiple E-PEs / ENNIs (as shown in figure above) where a given vES can be multi-homed to two or more PE devices (on two or more ENNIs) via their associated EVCs. Just like physical ES's in [EVPN] and [PBB-EVPN] solutions, these vES's can be single-homed or multi-homed ES's and when multi-homed, then can operate in either Single-Active or All-Active redundancy modes. In a typical SP off-network scenario, an ENNI can be associated with several thousands of single-homed vES's, several hundreds of Single-Active vES's and it may also be associated with tens or hundreds of All-Active vES's.

1.2 Virtual Ethernet Segments in Access MPLS Networks

Other Service Providers (SPs) want to extend the concept of the physical links in an ES to individual Pseudowires (PW) or to MPLS Label Switched Paths (LSPs) per [EVPN-VPWS] in Access MPLS networks. Figure 2 illustrates this concept.

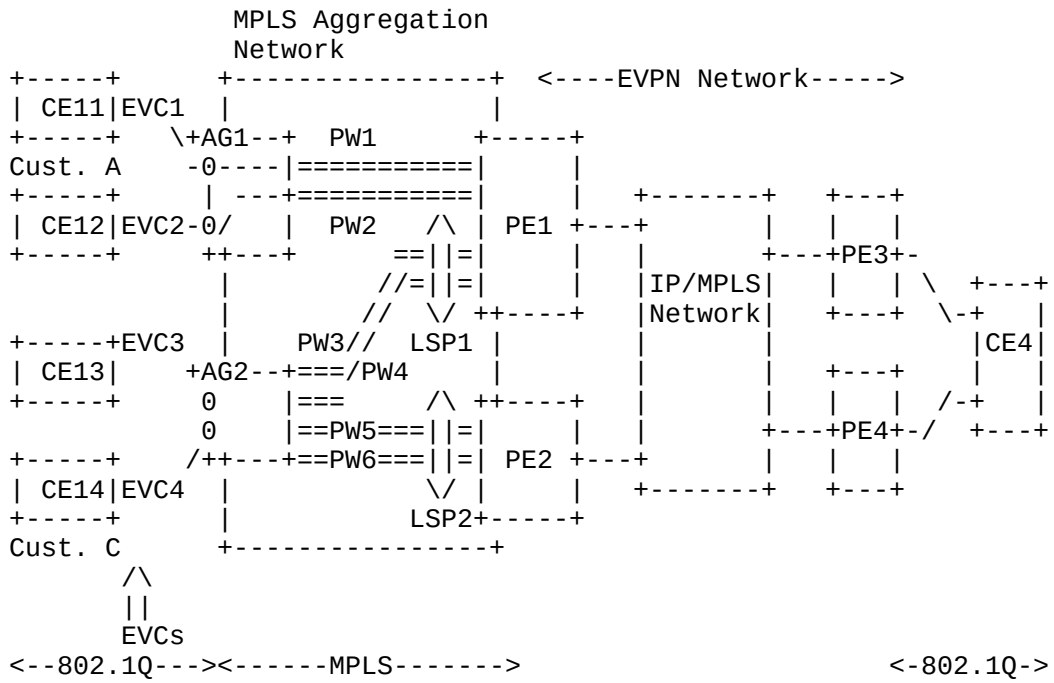


Figure 2: DHN and SH on Access MPLS networks

In some cases, Service Providers use Access MPLS Networks that belong to separate administrative entities or third parties as a way to get access to the their own IP/MPLS network infrastructure. This is the case illustrated in Figure 2.

An ES is defined as a set of individual PWs if they cannot be aggregated into a common LSP. If the aggregation of PWs is possible, the ES can be associated to an LSP in a given PE. In the example of Figure 2, EVC3 is connected to a VPWS instance in AG2 that is connected to PE1 and PE2 via PW3 and PW5 respectively. EVC4 is connected to a separate VPWS instance on AG2 that gets connected to

an EVI on PE1 and PE2 via PW4 and PW6, respectively. Since the PWs for the two VPWS instances can be aggregated into the same LSPs going to the EVPN network, a common virtual ES can be defined for LSP1 and LSP2. This ES will be shared by two separate EVIs in the EVPN network.

In some cases, this aggregation of PWs into common LSPs may not be possible. For instance, if PW3 were terminated into a third PE, e.g. PE3, instead of PE1, the ES would need to be defined on a per individual PW on each PE, i.e. PW3 and PW5 would belong to ES-1, whereas PW4 and PW6 would be associated to ES-2.

An ES that consists of a set of LSPs or individual PWs is also referred as virtual ES (vES) in this document."

This draft describes requirements and the extensions needed to support vES in [EVPN] and [PBB-EVPN]. Section 3 lists the set of requirements for Virtual ES's. Section 4 describes the solution for [PBB-EVPN] to meet these requirements. Section 5 describes the failure handling and recovery for Virtual ES's in [PBB-EVPN]. Section 6 covers scalability and fast convergence required for Virtual ES's in [PBB-EVPN].

2. Terminology

AC: Attachment Circuit
BEB: Backbone Edge Bridge
B-MAC: Backbone MAC Address
CE: Customer Edge
CFM: Connectivity Fault Management
C-MAC: Customer/Client MAC Address
DHD: Dual-homed Device
DHN: Dual-homed Network
ENNI: External Network-Network Interface
ES: Ethernet Segment
ESI: Ethernet-Segment Identifier
EVC: Ethernet Virtual Circuit
EVPN: Ethernet VPN
LACP: Link Aggregation Control Protocol
PE: Provider Edge
SH: Single-Homed

Single-Active Redundancy Mode (SA): 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 (AA): 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.

3. Requirements

This section describes the requirements specific to virtual Ethernet Segment (vES) for (PBB-)EVPN solutions. These requirements are in addition to the ones described in [EVPN-REQ], [EVPN], and [PBB-EVPN].

3.1. Single-Homed & Multi-Homed Virtual Ethernet Segments

A PE needs to support the following types of vES's:

(R1a) A PE MUST handle single-homed vES's on a single physical port (e.g., single ENNI)

(R1b) A PE MUST handle a mix of Single-Homed vES's and Single-Active multi-homed vES's simultaneously on a single physical port (e.g., single ENNI). Single-Active multi-homed vES's will be simply referred to as Single-Active vES's through the rest of this document.

(R1c) A PE MAY handle All-Active multi-homed vES's on a single physical port. All-Active multi-homed vES's will be simply referred to as All-Active vES's through the rest of this document.

(R1d) A PE MAY handle a mixed of All-Active vES's along with other types of vES's on a single physical port

(R1e) A Multi-Homed vES (Single-Active or All-Active) can be spread across any two or more PEs (on two or more ENNIs)

3.2. Scalability

A single physical port (e.g., ENNI) can be associated with many vES's. The following requirements give a quantitative measure for each vES type.

(R2a) A PE MUST handle thousands or tens of thousands of Single-homed vES's on a single physical port (e.g., single ENNI)

(R2b) A PE MUST handle hundreds of Single-Active vES's on a single physical port (e.g., single ENNI)

(R2c) A PE MAY handle tens or hundreds of All-Active Multi-Homed

vES's on a single physical port (e.g., single ENNI)

(R2d) A PE MUST handle the above scale for a mix of Single-homed vES's and Single-Active vES's simultaneously on a single physical port (e.g., single ENNI)

(R4e) A PE MAY handle the above sale for a mixed of All-Active Multi-Homed vES's along with other types of vES's on a single physical port

3.3. Local Switching

Many vES's of different types can be aggregated on a single physical port on a PE device and some of these vES can belong to the same service instance (or customer). This translates into the need for supporting local switching among the vES's of the same service instance on the same physical port (e.g., ENNI) of the PE.

(R3a) A PE MUST support local switching among different vES's belonging to the same service instance (or customer) on a single physical port. For example, in the above figure (1), PE1 MUST support local switching between CE11 and CE12 (both belonging to customer A) that are mapped to two Single-homed vES's on ENNI1.

In case of Single-Active vES's, the local switching is performed among active EVCs belonging to the same service instance on the same ENNI.

3.4. EVC Service Types

A physical port (e.g., ENNI) of a PE can aggregate many EVCs each of which is associated with a vES. Furthermore, an EVC may carry one or more VLANs. Typically, an EVC carries a single VLAN and thus it is associated with a single broadcast domain. However, there is no restriction on an EVC to carry more than one VLANs.

(R4a) An EVC can be associated with a single broadcast domain - e.g., VLAN-based service or VLAN bundle service

(R4b) An EVC MAY be associated with several broadcast domains - e.g., VLAN-aware bundle service

In the same way, a PE can aggregated many LSPs and PWs. In the case of individual PWs per vES, typically a PW is associated with a single broadcast domain, but there is no restriction on the PW to carry more than one VLAN if the PW is defined as vc-type VLAN.

(R4c) A PW can be associated with a single broadcast domain - e.g., VLAN-based service or VLAN bundle service.

(R4b) An PW MAY be associated with several broadcast domains - e.g., VLAN-aware bundle service."

3.5. Designated Forwarder (DF) Election

Section 8.5 of [EVPN] describes the default procedure for DF election in EVPN which is also used in [PBB-EVPN]. This default DF election procedure is performed at the granularity of <ESI, EVI>. In case of a vES, the same EVPN default procedure for DF election also applies; however, at the granularity of <vESI, EVI>; where vESI is the virtual Ethernet Segment Identifier. As in [EVPN], this default procedure for DF election at the granularity of <vESI, EVI> is also referred to as "service carving"; where, EVI is represented by an I-SID in PBB-EVPN and by a EVI service-id/vpn-id in EVPN. With service carving, it is possible to evenly distribute the DFs for different vES's among different PEs, thus distributing the traffic among different PEs. The following list the requirements apply to DF election of vES's for EVPN.

(R5a) A vES with m EVCs can be distributed among n ENNIs belonging to p PEs in any arbitrary oder; where n >= P >= m. For example, if there is an vES with 2 EVCs and there are 5 ENNIs on 5 PEs (PE1 through PE5), then vES can be dual-homed to PE2 and PE4 and the DF election must be performed between PE2 and PE4.

(R5b) Each vES MUST be identified by its own virtual ESI (vESI)

3.6. OAM

In order to detect the failure of individual EVC and perform DF election for its associated vES as the result of this failure, each EVC should be monitored independently.

(R6a) Each EVC SHOULD be monitored for its health independently

(R6b) A single EVC failure (among many aggregated on a single physical port/ENNI) MUST trigger DF election for its associated vES.

3.7. Failure & Recovery

(R7a) Failure and failure recovery of an EVC for a Single-homed vES SHALL NOT impact any other EVCs for its own service instance or any other service instances. In other words, for PBB-EVPN, it SHALL NOT trigger any MAC flushing both within its own I-SID as well as other I-SIDs.

(R7b) In case of All-Active Multi-Homed vES, failure and failure

recovery of an EVC for that vES SHALL NOT impact any other EVCs for its own service instance or any other service instances. In other words, for PBB-EVPN, it SHALL NOT trigger any MAC flushing both within its own I-SID as well as other I-SIDs.

(R7c) Failure & failure recovery of an EVC for a Single-Active vES SHALL only impact its own service instance. In other words, for PBB-EVPN, MAC flushing SHALL be limited to the associated I-SID only and SHALL NOT impact any other I-SIDs.

(R7d) Failure & failure recovery of an EVC for a Single-Active vES MAY only impact C-MACs associated with MHD/MHNs for that service instance. In other words, MAC flushing SHOULD be limited to single service instance (I-SID in the case of PBB-EVPN) and only CMACs for Single-Active MHD/MHNs.

3.8. Fast Convergence

Since large number of EVCs (and their associated vES's) are aggregated via a single physical port (e.g., ENNI), then the failure of that physical port impacts large number of vES's and triggers large number of ES route withdrawals. Formulating, sending, receiving, and processing such large number of BGP messages can introduce delay in DF election and convergence time. As such, it is highly desirable to have a mass-withdraw mechanism similar to the one in the [EVPN] for withdrawing large number of Ethernet A-D routes.

(R8a) There SHOULD be a mechanism equivalent to EVPN mass-withdraw such that upon an ENNI failure, only a single BGP message is needed to indicate to the remote PEs to trigger DF election for all impacted vES associated with that ENNI.

4. Solution Overview

The solutions described in [EVPN] and [PBB-EVPN] are leveraged as is with one simple modification and that is the ESI assignment is performed for a group of EVCs instead of a group of links. In other words, the ESI is associated with a virtual ES (vES) and that's why it will be referred to as vESI.

For EVPN solution, everything basically remains the same except for the handling of physical port failure where many vES's can be impacted. Section 5.1 and 5.3 below describe the handling of physical port/link failure for EVPN. In a typical multi-homed operation, MAC addresses are learned behind a vES are advertised with the ESI corresponding to the vES (i.e., vESI). EVPN aliasing and mass-withdraw operations are performed with respect to vES. In other

words, the Ethernet A-D routes for these operations are advertised with vESI instead of ESI.

For PBB-EVPN solution, the main change is with respect to the BMAC address assignment which is performed similar to what is described in section 7.2.1.1 of [PBB-EVPN] with the following refinements:

- One shared BMAC address is used per PE for the single-homed vES's. In other words, a single BMAC is shared for all single-homed vES's on that PE.

- One shared BMAC address should be used per PE per physical port (e.g., ENNI) for the Single-Active vES's. In other words, a single BMAC is shared for all Single-Active vES's that shared the same ENNI.

- One shared BMAC address can be used for all Single-Active vES's on that PE.

- One BMAC address is used per EVC per physical port per PE for each All-Active multi-homed vES. In other words, a single BMAC address is used per vES for All-Active multi-homing scenarios.

- A single BMAC address may also be used per vES per PE for Single-Active multi-homing scenarios.

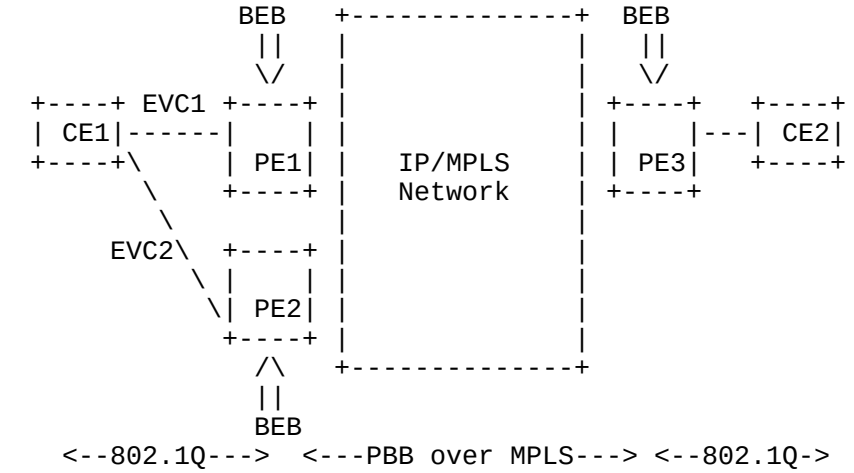


Figure 2: PBB-EVPN Network

4.1. EVPN DF Election for vES

The procedure for service carving for virtual Ethernet Segments is the same as the one outlined in section 8.5 of [EVPN] except for the fact that ES is replaced with vES. For the sake of clarity and completeness, this procedure is repeated below:

1. When a PE discovers the ESI or is configured with the ESI associated with its attached vES, it advertises an Ethernet Segment route with the associated ES-Import extended community attribute.

2. The PE then starts a timer (default value = 3 seconds) to allow the reception of Ethernet Segment routes from other PE nodes connected to the same vES. This timer value MUST be same across all PEs connected to the same vES.

3. When the timer expires, each PE builds an ordered list of the IP addresses of all the PE nodes connected to the vES (including itself), in increasing numeric value. Each IP address in this list is extracted from the "Originator Router's IP address" field of the advertised Ethernet Segment route. Every PE is then given an ordinal indicating its position in the ordered list, starting with 0 as the ordinal for the PE with the numerically lowest IP address. The ordinals are used to determine which PE node will be the DF for a given EVPN instance on the vES using the following rule: Assuming a redundancy group of N PE nodes, the PE with ordinal i is the DF for an EVPN instance with an associated EVI ID value of V when (V mod N) = i.

It should be noted that using "Originator Router's IP address" field in the Ethernet Segment route to get the PE IP address needed for the ordered list, allows for a CE to be multi-homed across different ASes if such need ever arises.

4. The PE that is elected as a DF for a given EVPN instance will unblock traffic for that EVPN instance. Note that the DF PE unblocks all traffic in both ingress and egress directions for Single-Active vES and unblocks multi-destination in egress direction for All-Active Multi-homed vES. All non-DF PEs block all traffic in both ingress and egress directions for Single-Active vES and block multi-destination traffic in the egress direction for All-Active multi-homed vES.

In the case of an EVC failure, the affected PE withdraws its Ethernet Segment route. This will re-trigger the service carving procedures on all the PEs in the RG. For PE node failure, or upon PE commissioning or decommissioning, the PEs re-trigger the service carving across all affected vES's. In case of a Single-Active multi-homing, when a service moves from one PE in the RG to another PE as a result of re-carving, the PE, which ends up being the elected DF for the service, SHOULD trigger a MAC address flush notification towards the

associated vES. This can be done, for e.g. using IEEE 802.1ak MVRP 'new' declaration.

For LSP and PW based vES, the non-DF PE SHOULD signal PW-status 'standby' signaling to the AG PE, and the new DF MAY send an LDP MAC withdraw message as a MAC address flush notification.

5. Failure Handling & Recovery

There are a number of failure scenarios to consider such as:

A: CE Uplink Port Failure
B: Ethernet Access Network Failure
C: PE Access-facing Port or link Failure
D: PE Node Failure
E: PE isolation from IP/MPLS network

[EVPN] and [PBB-EVPN] solutions provide protection against such failures as described in the corresponding references. In the presence of virtual Ethernet Segments (vES's) in these solutions, besides the above failure scenarios, there is one more scenario to consider and that is EVC failure. This implies that individual EVCs need to be monitored and upon their failure detection, appropriate DF election procedures and failure recovery mechanism need to be executed.

[ETH-OAM] is used for monitoring EVCs and upon failure detection of a given EVC, DF election procedure per section [4.1] is executed. For PBB-EVPN, some addition extensions are needed to failure handling and recovery procedures of [PBB-EVPN] in order to meet the above requirements. These extensions are describe in the next section.

[MPLS-OAM] and [PW-OAM] are used for monitoring the status of LSPs and/or PWs associated to vES.

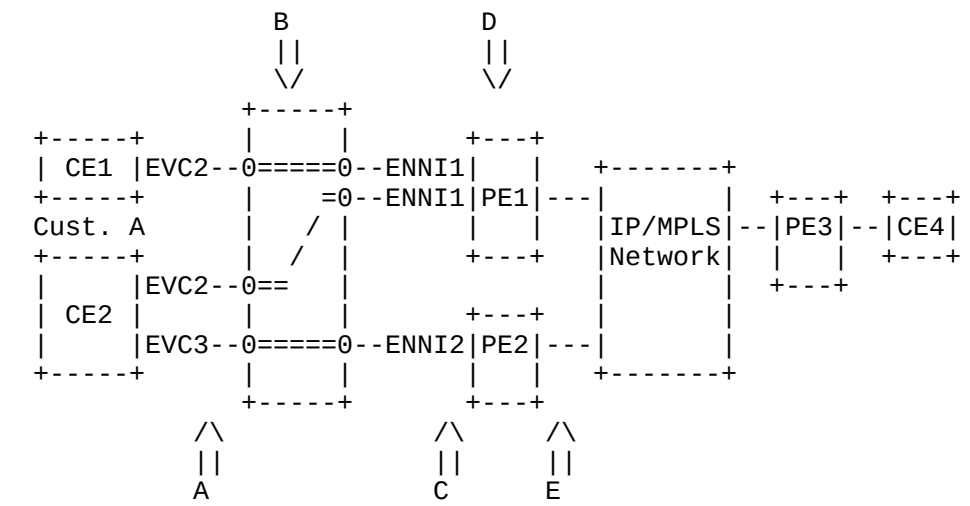


Figure 3: Failure Scenarios A,B,C,D and E

5.1. Failure Handling for Single-Active vES in EVPN

When a PE connected to a Single-Active multi-homed Ethernet Segment loses connectivity to the segment, due to link or port failure, it signals the remote PE to flush all CMAC addresses associated with that Ethernet Segment. This is done by advertising a mass-withdraw message using Ethernet A-D per-ES route. To be precise, there is no MAC flush per-se if there is only one backup PE for a given ES - i.e., only an update of the forwarding entries per backup-path procedure in [RFC 7432].

In case of an EVC failure that impacts a single vES, the exact same EVPN procedure is used. In this case, the message using Ethernet A-D per ES route carries the vESI representing the vES which is in turn associated with the failed EVC. The remote PEs upon receiving this message perform the same procedures outlined in section 8.2 of [EVPN].

5.2. EVC Failure Handling for Single-Active vES in PBB-EVPN

When a PE connected to a Single-Active multi-homed Ethernet Segment loses connectivity to the segment, due to link or port failure, it signals the remote PE to flush all CMAC addresses associated with that Ethernet Segment. This is done by advertising a BMAC route along with MAC Mobility Extended community.

In case of an EVC failure that impacts a single vES, if the above

PBB-EVPN procedure is used, it results in excessive CMAC flushing because a single physical port can support large number of EVCs (and their associated vES's) and thus advertising a BMAC corresponding to the physical port with MAC mobility Extended community will result in flushing CMAC addresses not just for the impacted EVC but for all other EVCs on that port.

In order to reduce the scope of CMAC flushing to only the impacted service instances (the service instance(s) impacted by the EVC failure), the BGP flush message is sent along with a list of impacted I-SID(s) represented by the new EVPN I-SID Extended Community as defined in section 6. Since typically an EVC maps to a single broadcast domain and thus a single service instance, the list only contains a single I-SID. However, if the failed EVC carries multiple VLANs each with its own broadcast domain, then the list contains several I-SIDs - one for each broadcast domain. This new BGP flush message basically instructs the remote PE to perform flushing for CMACs corresponding to the advertised BMAC only across the advertised list of I-ISIDs (which is typically one).

The above BMAC route that is advertised with the MAC Mobility Extended Community, can either represent the MAC address of the physical port that the failed EVC is associated with, or it can represent the MAC address of the PE. In the latter case, this is the dedicated MAC address used for all Single-Active vES's on that PE. The former one performs better than the latter one in terms of reducing the scope of flushing as described below and thus it is the recommended approach.

Advertising the BMAC route that represent the physical port (e.g., ENNI) on which the failed EVC reside along with MAC Mobility and I-SID extended communities provide the most optimum mechanism for CMAC flushing upon EVC failure in PBB-EVPN for Single-Active vES because:

1) Only CMAC addresses for the impacted service instances are flushed.

2) Only a subset of CMAC addresses for the impacted service instances are flushed - only the ones that are learned over the BMAC associated with the failed EVC. In other words, only a small fraction of the CMACs for the impacted service instance(s) are flushed.

5.3. Port Failure Handling for Single-Active vES's in EVPN

When a large number of EVCs are aggregated via a single physical port on a PE; where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vES's. If the

number of EVCs corresponding to the Single-Active vES’s for that physical port is in thousands, then thousands of service instances are impacted. Therefore, the BGP flush message need to be inclusive of all these impacted service instances. In order to achieve this, the following extensions are added to the baseline EVPN mechanism:

1) A PE when advertises an Ether-AD per ES route for a given vES, it colors it with the MAC address of the physical port which is associated with that vES. The receiving PEs take note of this color and create a list of vES’s for this color.

2) Upon a port failure (e.g., ENNI failure), the PE advertise a special mass-withdraw message with the MAC address of the failed port (i.e., the color of the port) encoded in the ESI field. For this encoding, type 3 ESI is used with the MAC field set to the MAC address of the port and the 3-octet local discriminator field set to 0xFFFFFF. This mass-withdraw route is advertised with a list of Route Targets corresponding to the impacted service instances. If the number of Route Targets is more than they can fit into a single attribute, then a set of Ethernet A-D per ES routes are advertised. The remote PEs upon receiving this message, realize that this is a special mass-withdraw message and they access the list of the vES’s for the specified color. Next, they initiate mass-withdraw procedure for each of the vES’s in the list.

5.4. Port Failure Handling for Single-Active vES’s in PBB-EVPN

When a large number of EVCs are aggregated via a single physical port on a PE; where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vES’s. If the number of EVCs corresponding to the Single-Active vES’s for that physical port is in thousands, then thousands of service instances (I-SIDs) are impacted. Therefore, the BGP flush message need to be sent with a list of thousands of I-SIDs. The new I-SID Extended Community provides a way to encode upto 24 I-SIDs in each Extended Community if the impacted I-SIDs are sequential (the base I-SID value plus the next 23 I-SID values). So, the packing efficiency can range from 1 to 24 and there can be up to 400 such Extended Community sent along with a BGP flush message for a total of 400 to 9600 I-SIDs. If the number of I-SIDs is large enough to not fit in a single Attribute, then either a number of BGP flush messages (with different RDs) can be transmitted or a single BGP flush message without the I-SID list can be transmitted. If the BGP flush message is transmitted without the I-SID list, then it instructs the receiving PEs to flush CMACs associated with that BMAC across all I-SIDs. For simplicity, we opt for the latter option in this document. In other words, if the number of impacted I-SIDs exceed that of a single BGP flush message,

then the flush message is sent without the I-SID list.

As also described in [PBB-EVPN], there are two ways to signal flush message upon a physical port failure:

1) If the MAC address of the physical port is used for PBB encapsulation as BMAC SA, then upon the port failure, the PE MUST use the EVPN MAC route withdrawal message to signal the flush

2) If the PE shared MAC address is used for PBB encapsulation as BMAC SA, then upon the port failure, the PE MUST re-advertise this MAC route with the MAC Mobility Extended Community to signal the flush

The first method is recommended because it reduces the scope of flushing the most.

5.5. Fast Convergence in PBB-EVPN

As described above, when a large number of EVCs are aggregated via a physical port on a PE; where each EVC corresponds to a vES, then the port failure impacts all the associated EVCs and their corresponding vES's. Two actions must be taken as the result of such port failure:

- Flushing of all CMACs associated with the BMAC of the failed port for the impacted I-SIDs

- DF election for all impacted vES's associated with the failed port

Section 5.4 describes how to flush CMAC address in the most optimum way - e.g., to flush least number of CMAC addresses for the impacted I-SIDs. This section describes how to perform DF election in the most optimum way - e.g., to trigger DF election for all impacted vES's (which can be in thousands) among the participating PEs via a single BGP message as opposed to sending thousands of BGP messages - one per vES.

In order to devise such fast convergence mechanism that can be triggered via a single BGP message, all vES's associated with a given physical port (e.g., ENNI) are colored with the same color representing that physical port. The MAC address of the physical port is used for this coloring purposes and when the PE advertises an ES route for a vES associated with that physical port, it advertises it with an EVPN MAC Extended Community indicating the color of that port.

The receiving PEs take note of this color and for each such color,

they create a list of vES's associated with this color (with this MAC address). Now, when a port failure occurs, the impacted PE needs to notify the other PEs of this color so that these PEs can identify all the impacted vES's associated with that color (from the above list) and re-execute DF election procedures for all the impacted vES's.

In PBB-EVPN, there are two ways to convey this color to other PEs upon a port failure - one corresponding to each method for signaling flush message as described in section 5.4. If for PBB encapsulation, the MAC address of the physical port is used as BMAC SA, then upon the port failure, the PE sends MAC withdrawal message with the MAC address of the failed port as the color. However, if for PBB encapsulation, the shared MAC address of the PE (dedicated for all Single-Active vES's) is used as BMAC SA, then upon the port failure, the PE re-advertises the MAC route (that carries the shared BMAC) along with this new EVPN MAC Extended Community to indicate the color along with MAC Mobility Extended Community.

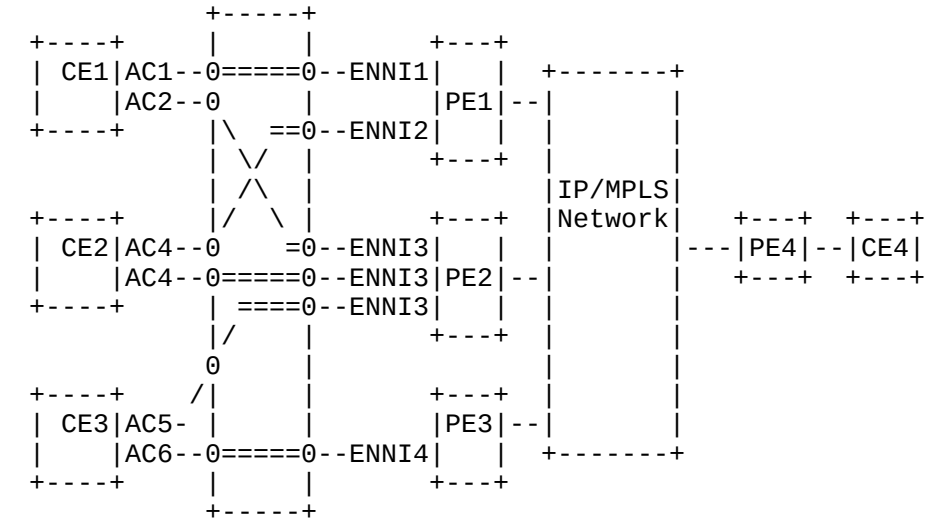


Figure 4: Fast Convergence Upon ENNI Failure

The following describes the procedure for coloring vES's and fast convergence using this color in more details:

1- When a vES is configured, the PE colors the vES with the MAC address of the corresponding physical port and advertises the Ethernet Segment route for this vES with this color.

2- All other PEs (in the redundancy group) take note of this color and add the vES to the list for this color.

3- Upon the occurrence of a port failure (e.g., an ENNI failure), the PE sends the flush message in one of the two ways described above indicating this color.

4- On reception of the flush message, other PEs use this info to flush their impacted CMACs and to initiate DF election procedures across all their affected vES's.

5- The PE with the physical port failure (ENNI failure), also send ES route withdrawal for every impacted vES's. The other PEs upon receiving these messages, clear up their BGP tables. It should be noted the ES route withdrawal messages are not used for executing DF election procedures by the receiving PEs.

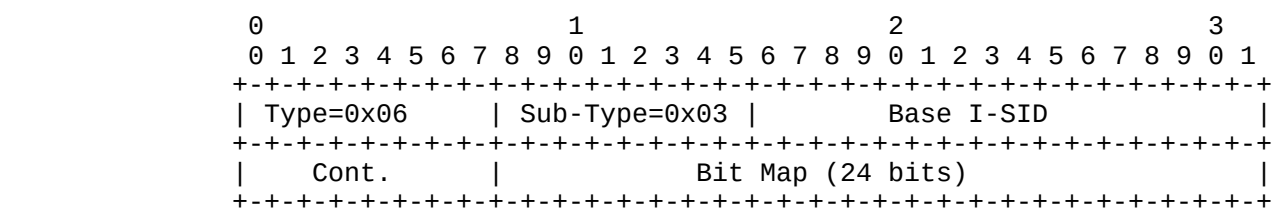
6. BGP Encoding

This document defines one new BGP Extended Community for EVPN.

6.1. I-SID Extended Community

A new EVPN BGP Extended Community called I-SID is introduced. This new extended community is a transitive extended community with the Type field of 0x06 (EVPN) and the Sub-Type of 0x04.

The I-SID Extended Community is encoded as an 8-octet value as follows:



This extended community is used to indicate the list of I-SIDs associated with a given Ethernet Segment.

24-bit map represents the next 24 I-SID after the base I-SID. For example based I-SID of 10025 with 24-bit map of zero means, only a single I-SID of 10025. I-SID of 10025 with bit map of 0x000001 means there are two I-SIDs, 10025 and 10026.

7. Acknowledgements

TBD

8. Security Considerations This document does not introduce any additional security constraints.

9. IANA Considerations

TBD

10. Intellectual Property Considerations

This document is being submitted for use in IETF standards discussions.

11. Normative References

[PBB] Clauses 25 and 26 of "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q, 2013.

12. Informative References

[RFC7209] Sajassi, et al., "Requirements for Ethernet VPN (EVPN)", RFC7209, May 2014.

[EVPN] Sajassi, et al., "BGP MPLS Based Ethernet VPN", draft-ietf-l2vpn-evpn-07.txt, work in progress, May 7, 2014.

[PBB-EVPN] Sajassi, et al., "PBB-EVPN", draft-ietf-l2vpn-pbb-evpn-07.txt, work in progress, June 18, 2014.

13. Authors' Addresses

Ali Sajassi
Cisco Systems
Email: sajassi@cisco.com

Patrice Brissette
Cisco Systems
Email: pbrisset@cisco.com

Rick Schell
Verizon
Email: richard.schell@verizon.com

John E Drake
Juniper
Email: jdrake@juniper.net

Tapraj Singh
Juniper
Email: tsingh@juniper.net

Jorge Rabadan
ALU
Email: jorge.rabadan@alcatel-lucent.com

PALS Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 7, 2016

H. Shah
Ciena Corporation
P. Brissette
R. Rahman
K. Raza
Cisco Systems, Inc.
Z. Li
Z. Shunwan
W. Haibo
Huawei Technologies
I. Chen
Ericsson
M. Bocci
Alcatel-Lucent
J. Hardwick
Metaswitch
S. Esale
Junipr Networks
K. Tiruveedhula
T. Singh
Juniper Networks
I. Hussain
Infinera Corporation
B. Wen
J. Walker
Comcast
N. Delregno
L. Jalil
M. Joecylyn
Verizon

July 06, 2015

YANG Data Model for MPLS-based L2VPN
draft-shah-pals-mpls-l2vpn-yang-00.txt

Abstract

This document describes a YANG data model for Layer 2 VPN services over MPLS networks. These services include Virtual Private Wire Service (VPWS), Virtual Private LAN service (VPLS) and Ethernet Virtual Private Service (EVPN) that uses LDP and BGP signaled Pseudowires. This document mainly focuses on L2VPN VPWS, other services are for future investigations.

Status of This Memo

Copyright Notice

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 MPLS based Layer 2 VPN services (L2VPN) [RFC4664] as well as switching between the local attachment circuits. The L2VPN services include point-to-point VPWS and Multipoint VPLS and EVPN services. These services are realized by signaling Pseudowires across MPLS networks using LDP [RFC4447][RFC4762] or BGP[RFC4761].

The Yang data model in this document defines Ethernet based Layer 2 services. Other Layer 2 services, such as ATM, Frame Relay, TDM, etc are included in the scope but will be covered as the future work items. The Ethernet based Layer 2 services will leverage the definitions used in other standards organizations such as IEEE 802.1 and Metro Ethernet Forum (MEF).

The goal is to propose a data object model consisting of building blocks that can be assembled in different order to realize different services. The definition work is undertaken initially by a smaller working group with members representing various vendors and service providers. The VPWS service definitions are covered first followed by VPLS services that build on the data blocks defined for VPWS.

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

- Configuration
- Operational State
- Executables (Actions)
- Notifications

The document is organized to first define the data model for the configuration, operational state, actions and notifications of VPWS. The L2VPN data object model defined in this document uses the instance centric approach whereby VPWS service attributes are specified for a given VPWS instance.

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. L2VPN YANG Model

3.1. Overview

One single top level container, mpls-l2vpn, is defined as a parent for three different second level containers that are vpws-instances, vpls-instances, and common building blocks of AC-templates(Attachment Circuit templates) and pseudowire-templates. This document defines the vpws-instances and templates for AC and Pseudowires. The definition of vpls-instances and evpn-instances is left for future revisions.

The L2VPN services have been defined in the IETF L2VPN working group but leverages the pseudowire technologies that were defined in the PWE3 working group. A large number of RFCs from these working groups cover this subject matter. Hence, it is prudent that this document state the scope of the MPLS L2VPN object model definitions.

The following documents are within the scope. This is not an exhaustive list but a representation of documents that are covered for this work:

- Requirements for Pseudo-wire Emulation Edge-to-Edge (PWE3) [RFC3916]
- Pseudo-wire Emulation Edge-to-Edge (PWE3) Architecture [RFC3985]
- IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3) [RFC4446]
- Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP) [RFC4447]
- Encapsulation Methods for Transport of Ethernet over MPLS Networks [RFC4448]

- Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN [RFC4385]
- Requirements for Multi-Segment Pseudowire Emulation Edge-to-Edge (PWE3) [RFC5254]
- An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge [RFC5659]
- Segmented Pseudowire [RFC6073]
- Framework for Layer 2 Virtual Private Networks [RFC4664]
- Service Requirements for Layer 2 Provider-Provisioned Virtual Private Networks [RFC4665]
- Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling [RFC4761]
- Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling [RFC4762]
- Attachment Individual Identifier (AII) Types for Aggregation [RFC5003]
- Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs) [RFC6074]
- Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network [RFC6391]
- Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling [RFC6624]
- Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging [RFC7041]
- LDP Extensions for Optimized MAC Address Withdrawal in a Hierarchical Virtual Private LAN Service (H-VPLS) [RFC7361]
- Using the generic associated channel label for Pseudowire in the MPLS Transport Profile [RFC6423]
- Pseudowire status for static pseudowire [RFC6478]

Note that while pseudowire over MPLS-TP related work is in scope, the initial effort will only address definitions of object model for VPWS services that are commonly deployed.

The ietf work in L2VPN and PWE3 working group relating to L2TP, OAM, multicast (e.g. p2mp, etree, etc) and access specific protocols such as G.8032, MSTP, etc is out-of-scope for this document.

The following is the high level view of the L2VPN data model.

```
template-ref AC // AC
                   template
                   attributes

template-ref PW // PW
                   template
                   attributes

vpws-instance name // container

        svc-type

        // list of AC and PW being used
        AC-1 // container
                template-ref AC
                attribute-override
        PW-2 // container
                template-ref PW
                attribute-override
        PW-3 // container
                template-ref PW
                attribute-override

        // ONLY 2 endpoints!!!
        endpoint-A // container
                        AC-1 // reference

        endpoint-Z // container
                redundancy-grp // container
                        PW-2 // reference
                        PW-3 // reference
```

Figure 1

3.2. L2VPN Common

3.2.1. ac-templates

The ac-templates container contains a list of ac-template. Each ac-template defines a list of AC attributes that are part of native services but associated and processed within the context of L2VPN. For instance, Ethernet VLAN tag imposition, disposition and translation or CVID-bundling would be part of this template.

3.2.2. pw-templates

The pw-templates container contains a list of pw-template. Each pw-template defines a list of common pseudowire attributes such as PW MTU, control word support etc.

3.3. VPWS

3.3.1. ac list

Each VPWS instance defines a list of AC which are cross-connected by the service. Each entry of the AC consists of one ac-template with predefined attributes and values, but also defines attributes that override the attributes defined in referenced ac-template.

3.3.2. pw list

Each VPWS instance defines a list of PW which are cross-connected by the service. Each entry of the PW consists of one pw-template with pre-defined attributes and values, but also defines attributes that override those defined in referenced pw-template. No restrictions are placed on type of signaling (i.e. LDP or BGP) used for a given PW. It is entirely possible to define two PWs, one signaled by LDP and other by BGP.

3.3.3. redundancy-grp choice

The redundancy-grp is a generic redundancy construct which can hold primary and backup members of AC and PWs. This flexibility permits combinations of -

- primary and backup AC
- primary and backup PW
- primary AC and backup PW
- primary PW and backup AC

3.3.4. endpoint container

The endpoint container holds AC, PW or redundancy-grp references. The core aspect of endpoint container is its flexible personality based on what user decides to include in it. It is future-proofed with possible extensions that can be included in the endpoint container such as Integrated Route Bridging (IRB), PW Headend, Virtual Switch Instance, etc.

3.3.5. vpws-instances container

The vpws-instances container contains a list of vpws-instance. Each entry of the vpws-instance represents a layer-2 cross-connection of two endpoints. This model defines three possible types of endpoints, ac, pw, and redundancy-grp, and allows a vpws-instance to cross-connect any one type of endpoint to all other types of endpoint.

The augmentation of ietf-mpls-l2vpn module is TBD. All IP addresses defined in this module are currently scoped under global VRF/table.

```
module: ietf-mpls-l2vpn
+--rw mpls-l2vpn
   +--rw common
   |  +--rw pw-templates
   |  |  +--rw pw-template* [name]
   |  |     +--rw name              string
   |  |     +--rw mtu?              uint32
   |  |     +--rw cw-negotiation?   cw-negotiation-type
   |  |     +--rw tunnel-policy?    string
   |  +--rw ac-templates
   |     +--rw ac-template* [name]
   |        +--rw name    string
   +--rw vpws-instances
   |  +--rw vpws-instance* [instance-name]
   |     +--rw instance-name     string
   |     +--rw description?      string
   |     +--rw service-type?     l2vpn-service-type
   |     +--rw discovery-type?   l2vpn-discovery-type
   |     +--rw signaling-type    l2vpn-signaling-type
   |     +--rw bgp-parameters
   |     |  +--rw common
   |     |  |  +--rw route-distinguisher?   string
   |     |  |  +--rw vpn-targets* [rt-value]
   |     |  |     +--rw rt-value    string
   |     |  |     +--rw rt-type     bgp-rt-type
   |     |  +--rw discovery
   |     |  |  +--rw vpn-id?   string
```

```
|     |  +--rw signaling
|     |     +--rw site-id?      uint16
|     |     +--rw site-range?   uint16
|     +--rw pw* [name]
|     |  +--rw name              string
|     |  +--rw cw-negotiation?   cw-negotiation-type
|     |  +--rw template?         pw-template-ref
|     |  +--rw vccv-ability?     boolean
|     |  +--rw tunnel-policy?    string
|     |  +--rw request-vlanid?   uint16
|     |  +--rw vlan-tpid?        string
|     |  +--rw ttl?              uint8
|     |  +--rw (pw-type)?
|     |     +--:(ldp-pw)
|     |     |  +--rw peer-ip?          inet:ip-address
|     |     |  +--rw pw-id?            uint32
|     |     |  +--rw transmit-label?   uint32
|     |     |  +--rw receive-label?    uint32
|     |     |  +--rw icb?              boolean
|     |     +--:(bgp-pw)
|     |     |  +--rw remote-pe-id?     inet:ip-address
|     |     +--:(bgp-ad-pw)
|     |        +--rw remote-ve-id?     uint16
|     +--rw ac* [name]
|     |  +--rw name                       string
|     |  +--rw template?                  ac-template-ref
|     |  +--rw pipe-mode?                 enumeration
|     |  +--rw link-discovery-protocol?  link-discovery-protocol-type
|     +--rw endpoint-a
|     |  +--rw (ac-or-pw-or-redundancy-grp)?
|     |     +--:(ac)
|     |     |  +--rw ac?                -> ../../ac/name
|     |     +--:(pw)
|     |     |  +--rw pw?                -> ../../pw/name
|     |     +--:(redundancy-grp)
|     |     |  +--rw (primary)
|     |     |  |  +--:(primary-pw)
|     |     |  |  |  +--rw primary-pw?        -> ../../pw/name
|     |     |  |  +--:(primary-ac)
|     |     |  |     +--rw primary-ac?        -> ../../ac/name
|     |     |  +--rw (backup)
|     |     |  |  +--:(backup-pw)
|     |     |  |  |  +--rw backup-pw?         -> ../../pw/name
|     |     |  |  +--:(backup-ac)
|     |     |  |     +--rw backup-ac?         -> ../../ac/name
|     |     |  +--rw protection-mode?   enumeration
|     |     +--:(reroute-mode)
|     |     |  +--rw reroute-mode?      enumeration
```

```
|     |     +--:(reroute-delay)
|     |     |  +--rw reroute-delay?     uint16
|     |     +--:(dual-receive)
|     |     |  +--rw dual-receive?      boolean
|     |     +--:(revert)
|     |     |  +--rw revert?            boolean
|     |     +--:(revert-delay)
|     |        +--rw revert-delay?      uint16
|     +--rw endpoint-z
|        +--rw (ac-or-pw-or-redundancy-grp)?
|           +--:(ac)
|           |  +--rw ac?                -> ../../ac/name
|           +--:(pw)
|           |  +--rw pw?                -> ../../pw/name
|           +--:(redundancy-grp)
|           |  +--rw (primary)
|           |  |  +--:(primary-pw)
|           |  |  |  +--rw primary-pw?        -> ../../pw/name
|           |  |  +--:(primary-ac)
|           |  |     +--rw primary-ac?        -> ../../ac/name
|           |  +--rw (backup)
|           |  |  +--:(backup-pw)
|           |  |  |  +--rw backup-pw?         -> ../../pw/name
|           |  |  +--:(backup-ac)
|           |  |     +--rw backup-ac?         -> ../../ac/name
|           |  +--rw protection-mode?   enumeration
|           +--:(reroute-mode)
|           |  +--rw reroute-mode?      enumeration
|           +--:(reroute-delay)
|           |  +--rw reroute-delay?     uint16
|           +--:(dual-receive)
|           |  +--rw dual-receive?      boolean
|           +--:(revert)
|           |  +--rw revert?            boolean
|           +--:(revert-delay)
|              +--rw revert-delay?      uint16
+--rw vpls-instances
```

Figure 2

4. YANG Module

The L2VPN configuration container is logically divided into following high level config areas:

```
<CODE BEGINS> file "ietf-mpls-l2vpn@2015-06-30.yang"
module ietf-mpls-l2vpn {
  namespace "urn:ietf:params:xml:ns:yang:ietf-mpls-l2vpn";
  prefix "mpls-l2vpn";

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

  organization  "ietf";
  contact       "ietf";
  description   "mpls-l2vpn";
  revision "2015-06-30" {
    description "Initial revision";
    reference   "";
  }

  /* identities */

  identity link-discovery-protocol {
    description "Base identiy from which identities describing " +
                "link discovery protocols are derived.";
  }

  identity lacp {
    base "link-discovery-protocol";
    description "This identity represents LACP";
  }

  identity lldp {
    base "link-discovery-protocol";
    description "This identity represents LLDP";
  }

  identity bpdu {
    base "link-discovery-protocol";
    description "This identity represens BPDU";
  }

  identity cpd {
    base "link-discovery-protocol";
    description "This identity represents CPD";
  }

  identity udld {
    base "link-discovery-protocol";
    description "This identity represens UDLD";
  }
```

```
     /* typedefs */

     typedef l2vpn-service-type {
       type enumeration {
         enum ethernet {
           description "Ethernet service";
         }
         enum ATM {
           description "Asynchronous Transfer Mode";
         }
         enum FR {
           description "Frame-Relay";
         }
         enum TDM {
           description "Time Division Multiplexing";
         }
       }
       description "L2VPN service type";
     }

     typedef l2vpn-discovery-type {
       type enumeration {
         enum manual {
           description "Manual configuration";
         }
         enum bgp-ad {
           description "Border Gateway Protocol (BGP) auto-discovery";
         }
       }
       description "L2VPN discovery type";
     }

     typedef l2vpn-signaling-type {
       type enumeration {
         enum static {
           description "Static configuration of labels (no signaling)";
         }
         enum ldp {
           description "Label Distribution Protocol (LDP) signaling";
         }
         enum bgp {
           description "Border Gateway Protocol (BGP) signaling";
         }
       }
       description "L2VPN signaling type";
     }

     typedef bgp-rt-type {
```

```
      type enumeration {
        enum import {
          description "For import";
        }
        enum export {
          description "For export";
        }
        enum both {
          description "For both import and export";
        }
      }
      description "BGP route-target type. Import from BGP YANG";
    }

    typedef cw-negotiation-type {
      type enumeration {
        enum "non-preferred" {
          description "No preference for control-word";
        }
        enum "preferred" {
          description "Prefer to have control-word negotiation";
        }
      }
      description "control-word negotiation preference type";
    }

    typedef link-discovery-protocol-type {
      type identityref {
        base "link-discovery-protocol";
      }
      description "This type is used to identify " +
                  "link discovery protocol";
    }

    typedef pw-template-ref {
      type leafref {
        path "/l2vpn/common/pw-templates/pw-template/name";
      }
      description "pw-template-ref";
    }

    typedef ac-template-ref {
      type leafref {
        path "/l2vpn/common/ac-templates/ac-template/name";
      }
      description "ac-tempalte-ref";
    }
```

```
     /* groupings */

     grouping vpws-endpoint {
       description
         "A vpws-endpoing could either be an ac or a pw";
       choice ac-or-pw-or-redundancy-grp {
         description "A choice ofattachment circuit or " +
                     "pseudowire or redundancy group";
         case ac {
           leaf ac {
             type leafref {
               path "../../ac/name";
             }
             description "reference to an attachment circuit";
           }
         }
         case pw {
           leaf pw {
             type leafref {
               path "../../pw/name";
             }
             description "reference to a pseudowire";
           }
         }
         case redundancy-grp {
           choice primary {
             mandatory true;
             description "primary options";
             case primary-pw  {
               leaf primary-pw {
                 type leafref {
                   path "../../pw/name";
                 }
                 description "primary pseudowire";
               }
             }
             case primary-ac {
               leaf primary-ac {
                 type leafref {
                   path "../../ac/name";
                 }
                 description "primary attachment circuit";
               }
             }
           }
           choice backup {
             mandatory true;
             description "backup options";
```

```
            case backup-pw {
              leaf backup-pw {
                type leafref {
                  path "../../pw/name";
                }
                description "backup pseudowire";
              }
            }
            case backup-ac {
              leaf backup-ac {
                type leafref {
                  path "../../ac/name";
                }
                description "backup attachment circuit";
              }
            }
          }
          leaf protection-mode {
            type enumeration {
              enum "frr" {
                value 0;
                description "fast reroute";
              }
              enum "master-slave" {
                value 1;
                description "master-slave";
              }
              enum "independent" {
                value 2;
                description "independent";
              }
            }
            description "protection-mode";
          }
        }
        leaf reroute-mode {
          type enumeration {
            enum "immediate" {
              value 0;
              description "immediate reroute";
            }
            enum "delayed" {
              value 1;
              description "delayed reroute";
            }
            enum "never" {
              value 2;
              description "never reroute";
```

```
            }
          }
          description "reroute-mode";
        }
        leaf reroute-delay {
          when "../reroute-mode = 'delayed'" {
            description
              "Specify amount of time to delay reroute " +
              "only when delayed route is configured";
          }
          type uint16;
          description
            "amount of time to delay reroute";
        }
        leaf dual-receive {
          type boolean;
          description
            "allow extra traffic to be carried by backup";
        }
        leaf revert {
          type boolean;
          description
            "allow forwarding to revert to primary " +
            "after restoring primary";
          /* This is called "revertive" during the discussion. */
        }
        leaf revert-delay {
          when "../revert = 'true'" {
            description
              "Specify the amount of time to wait to revert " +
              "to primary only if reversion is configured";
          }
          type uint16;
          description
            "amount ot time to wait to revert to primary";
          /* This is called "wtr" during discussion. */
        }
      }
    }

    /* We can define vpls-endpoing-grp that has the same structure as
     * vpws-endpoing-grp, but has more endpoint options.
     */

    /* L2VPN YANG Model */

    container l2vpn {
      description "l2vpn";
```

```
      container common {
        description "common l2pn attributes";
        container pw-templates {
          description "pw-templates";
          list pw-template {
            key "name";
            description "pw-template";
            leaf name {
              type string;
              description "name";
            }
            leaf mtu {
              type uint32;
              description "pseudowire mtu";
            }
            leaf cw-negotiation {
              type cw-negotiation-type;
              default "preferred";
              description
                "control-word negotiation preference";
            }
            leaf tunnel-policy {
              type string;
              description "tunnel policy name";
            }
          }
        }
        container ac-templates {
          description "attachment circuit templates";
          /* To be fleshed out in future revisions */
          list ac-template {
            key "name";
            description "ac-template";
            leaf name {
              type string;
              description "name";
            }
          }
        }
      }
      container vpws-instances {
        description "vpws-instances";
        list vpws-instance {
          key "instance-name";
          description "A VPWS instance";
          leaf instance-name {
            type string;
            description "Name of VPWS instance";
```

```
          }
          leaf description {
            type string;
            description "Description of the VPWS instance";
          }
          leaf service-type {
            type l2vpn-service-type;
            default ethernet;
            description "VPWS service type";
          }
          leaf discovery-type {
            type l2vpn-discovery-type;
            default manual;
            description "VPWS discovery type";
          }
          leaf signaling-type {
            type l2vpn-signaling-type;
            mandatory true;
            description "VPWS signaling type";
          }
          container bgp-parameters {
            description "Parameters for BGP";
            container common {
              when "../../discovery-type = 'bgp-ad'" {
                description "Check discovery type: " +
                            "Can only configure BGP discovery if " +
                            "discovery type is BGP-AD";
              }
              description "Common BGP parameters";
              leaf route-distinguisher {
                type string;
                description "BGP RD";
              }
              list vpn-targets {
                key rt-value;
                description "Route Targets";
                leaf rt-value {
                  type string;
                  description "Route-Target value";
                }
                leaf rt-type {
                  type bgp-rt-type;
                  mandatory true;
                  description "Type of RT";
                }
              }
            }
            container discovery {
```

```
            when "../../discovery-type = 'bgp-ad'" {
              description "BGP parameters for discovery: " +
                          "Can only configure BGP discovery if " +
                          "discovery type is BGP-AD";
            }
            description "BGP parameters for discovery";
            leaf vpn-id {
              type string;
              description "VPN ID";
            }
          }
          container signaling {
            when "../../signaling-type = 'bgp'" {
              description "Check signaling type: " +
                          "Can only configure BGP signaling if " +
                          "signaling type is BGP";
            }
            description "BGP parameters for signaling";
            leaf site-id {
              type uint16;
              description "Site ID";
            }
            leaf site-range {
              type uint16;
              description "Site Range";
            }
          }
        }
        list pw {
          key "name";
          description "pseudowire";
          leaf name {
            type string;
            description "pseudowire name";
          }
          leaf cw-negotiation {
            type cw-negotiation-type;
            default "preferred";
            description "Override the control-word negotiation " +
                        "preference specified in the " +
                        "pseudowire template.";
          }
          leaf template {
            type pw-template-ref;
            description "pseudowire template";
          }
          leaf vccv-ability {
            type boolean;
```

```
            description "vccvability";
          }
          leaf tunnel-policy {
            type string;
            description "Used to override the tunnel policy name " +
                        "specified in the pseduowire template";
          }
          leaf request-vlanid {
            type uint16;
            description "request vlanid";
          }
          leaf vlan-tpid {
            type string;
            description "vlan tpid";
          }
          leaf ttl {
            type uint8;
            description "time-to-live";
          }
          choice pw-type {
            description "A choice of pseudowire type";
            case ldp-pw {
              leaf peer-ip {
                type inet:ip-address;
                description "peer IP address";
              }
              leaf pw-id {
                type uint32;
                description "pseudowire id";
              }
              leaf transmit-label {
                type uint32;
                description "transmit lable";
              }
              leaf receive-label {
                type uint32;
                description "receive label";
              }
              leaf icb {
                type boolean;
                description "inter-chassis backup";
              }
            }
            case bgp-pw {
              leaf remote-pe-id {
                type inet:ip-address;
                description "remote pe id";
              }
```

```
            }
            case bgp-ad-pw {
              leaf remote-ve-id {
                type uint16;
                description "remote ve id";
              }
            }
          }
        }
        list ac {
          key "name";
          description "attachment circuit";
          leaf name {
            type string;
            description "name";
          }
          leaf template {
            type ac-template-ref;
            description "attachment circuit template";
          }
          leaf pipe-mode {
            type enumeration {
              enum "pipe" {
                value 0;
                description "regular pipe mode";
              }
              enum "short-pipe" {
                value 1;
                description "short pipe mode";
              }
              enum "uniform" {
                value 2;
                description "uniform pipe mode";
              }
            }
            description "pipe mode";
          }
          leaf link-discovery-protocol {
            type link-discovery-protocol-type;
            description "link discovery protocol";
          }
        }
        container endpoint-a {
          description "endpoint-a";
          uses vpws-endpoint;
        }
        container endpoint-z {
          description "endpoint-z";
```

```
            uses vpws-endpoint;
          }
        }
      }
      container vpls-instances {
        /* To be fleshed out in future revisions */
        description "vpls-instances";
      }
    }
  }

  <CODE ENDS>
```

Figure 3

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. Acknowledgments

The authors would like to acknowledge TBD for their useful comments.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

[RFC3916] Xiao, X., McPherson, D., and P. Pate, "Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)", RFC 3916, September 2004.

[RFC3985] Bryant, S. and P. Pate, "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, March 2005.

[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, February 2006.

[RFC4446] Martini, L., "IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)", BCP 116, RFC 4446, April 2006.

[RFC4447] Martini, L., Rosen, E., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", RFC 4447, April 2006.

[RFC4448] Martini, L., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, April 2006.

[RFC4664] Andersson, L. and E. Rosen, "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, September 2006.

[RFC4665] Augustyn, W. and Y. Serbest, "Service Requirements for Layer 2 Provider-Provisioned Virtual Private Networks", RFC 4665, September 2006.

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

[RFC4762] Lasserre, M. and V. Kompella, "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, January 2007.

[RFC5003] Metz, C., Martini, L., Balus, F., and J. Sugimoto, "Attachment Individual Identifier (AII) Types for Aggregation", RFC 5003, September 2007.

[RFC5254] Bitar, N., Bocci, M., and L. Martini, "Requirements for Multi-Segment Pseudowire Emulation Edge-to-Edge (PWE3)", RFC 5254, October 2008.

[RFC5659] Bocci, M. and S. Bryant, "An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge", RFC 5659, October 2009.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6073] Martini, L., Metz, C., Nadeau, T., Bocci, M., and M. Aissaoui, "Segmented Pseudowire", RFC 6073, January 2011.

[RFC6074] Rosen, E., Davie, B., Radoaca, V., and W. Luo, "Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)", RFC 6074, January 2011.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6391] Bryant, S., Filsfils, C., Drafz, U., Kompella, V., Regan, J., and S. Amante, "Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network", RFC 6391, November 2011.

[RFC6423] Li, H., Martini, L., He, J., and F. Huang, "Using the Generic Associated Channel Label for Pseudowire in the MPLS Transport Profile (MPLS-TP)", RFC 6423, November 2011.

[RFC6478] Martini, L., Swallow, G., Heron, G., and M. Bocci, "Pseudowire Status for Static Pseudowires", RFC 6478, May 2012.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFC6624] Kompella, K., Kothari, B., and R. Cherukuri, "Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling", RFC 6624, May 2012.

[RFC7041] Balus, F., Sajassi, A., and N. Bitar, "Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging", RFC 7041, November 2013.

[RFC7361] Dutta, P., Balus, F., Stokes, O., Calvignac, G., and D. Fedyk, "LDP Extensions for Optimized MAC Address Withdrawal in a Hierarchical Virtual Private LAN Service (H-VPLS)", RFC 7361, September 2014.

Authors' Addresses

Himanshu Shah
Ciena Corporation

Email: hshah@ciena.com

Patrice Brissette
Cisco Systems, Inc.

Email: pbrisset@cisco.com

Reshad Rahman
Cisco Systems, Inc.

Email: rrahman@cisco.com

Kamran Raza
Cisco Systems, Inc.

Email: skraza@cisco.com

Zhenbin Li
Huawei Technologies

Email: lizhenbin@huawei.com

Zhuang Shunwan
Huawei Technologies

Email: Zhuangshunwan@huawei.com

Wang Haibo
Huawei Technologies

Email: rainsword.wang@huawei.com

Ing-When Chen
Ericsson

Email: ing-wher.chen@ericsson.com

Mathew Bocci
Alcatel-Lucent

Email: mathew.bocci@alcatel-lucent.com

Jonathan Hardwick
Metaswitch

Email: jonathan.hardwick@metaswitch.com

Santosh Esale
Junipr Networks

Email: sesale@juniper.net

Kishore Tiruveedhula
Juniper Networks

Email: kishoret@juniper.net

Tapraj Singh
Juniper Networks

Email: tsingh@juniper.net

Iftekar Hussain
Infinera Corporation

Email: ihussain@infinera.com

Bin Wen
Comcast

Email: Bin_Wen@cable.comcast.com

Jason Walker
Comcast

Email: jason_walker2@cable.comcast.com

Nick Delregno
Verizon

Email: nick.deregno@verizon.com

Luay Jalil
Verizon

Email: luay.jalil@verizon.com

Maria Joecylyn
Verizon

Email: joecylyn.malit@verizon.com

BESS Workgroup
Internet Draft

Intended status: Informational

J. Rabadan, Ed.
S. Sathappan
K. Nagaraj
W. Henderickx
G. Hankins
Alcatel-Lucent

T. King
D. Melzer
DE-CIX

Expires: January 7, 2016 July 6, 2015

Operational Aspects of Proxy-ARP/ND in EVPN Networks
draft-snr-bess-evpn-proxy-arp-nd-01

Abstract

The MAC/IP Advertisement route specified in [RFC7432] can optionally carry IPv4 and IPv6 addresses associated with a MAC address. Remote PEs can use this information to reply locally (act as proxy) to IPv4 ARP requests and IPv6 Neighbor Solicitation messages and reduce/suppress the flooding produced by the Address Resolution procedure. This EVPN capability is extremely useful in Internet Exchange Points (IXPs) and Data Centers (DCs) with large broadcast domains, where the amount of ARP/ND flooded traffic causes issues on routers and CEs, as explained in [RFC6820]. This document describes how the [RFC7432] EVPN proxy-ARP/ND function may be implemented to help IXPs and other operators deal with the issues derived from Address Resolution in large broadcast domains.

Status of this Memo

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

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

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

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

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

This Internet-Draft will expire on January 7, 2015.

Table of Contents

1. Terminology

BUM: Broadcast, Unknown unicast and Multicast layer-2 traffic.

ARP: Address Resolution Protocol.

GARP: Gratuitous ARP message.

ND: Neighbor Discovery Protocol.

NS: Neighbor Solicitation message.

NA: Neighbor Advertisement.

IXP: Internet eXchange Point.

IXP-LAN: it refers to the IXP's large Broadcast Domain to where Internet routers are connected.

DC: Data Center.

IP->MAC: it refers to an IP address associated to a MAC address. The entries may be of three different types: dynamic, static or EVPN-learned.

SN-multicast address: Refers to the Solicited-Node IPv6 multicast address used by NS messages.

NUD: Neighbor Unreachability Detection, as per [RFC4861].

DAD: Duplicate Address Detection, as per [RFC4861].

SLLA: Source Link Layer Address, as per [RFC4861].

TLLA: Target Link Layer Address, as per [RFC4861].

R-bit: Router Flag in NA messages, as per [RFC4861].

O-bit: Override Flag in NA messages, as per [RFC4861].

S-bit: Solicited Flag in NA messages, as per [RFC4861].

RT2: EVPN Route type 2 or MAC/IP Advertisement route, as per [RFC7432].

MAC or IP DA: MAC or IP Destination Address.

MAC or IP SA: MAC or IP Source Address.

AS-MAC: Anti-spoofing MAC.

2. Introduction

As specified in [RFC7432] the IP Address field in the MAC/IP Advertisement route may optionally carry one of the IP addresses associated with the MAC address. A PE may learn local IP->MAC pairs and advertise them in EVPN MAC/IP routes. The remote PEs may add those IP->MAC pairs to their Proxy-ARP/ND tables and reply to local ARP requests or Neighbor Solicitations, reducing and even suppressing in some cases the flooding in the EVPN network.

EVPN and its associated Proxy-ARP/ND function are extremely useful in Data Centers (DCs) or Internet Exchange Points (IXPs) with large broadcast domains, where the amount of ARP/ND flooded traffic causes issues on routers and CEs. [RFC6820] describes the Address Resolution problems in Large Data Center networks.

This document describes how the [RFC7432] proxy-ARP/ND function may be implemented to help IXPs, DCs and other operators deal with the issues derived from Address Resolution in large broadcast domains.

2.1. The DC Use-Case

As described in [RFC6820] the IPv4 and IPv6 Address Resolution can create a lot of issues in large DCs. The amount of flooding that Address Resolution creates, as well as other associated issues can be mitigated with the use of EVPN and its proxy-ARP/ND function.

2.2. The IXP Use-Case

The implementation described in this document is especially useful in IXP networks.

A typical IXP provides access to a large layer-2 peering network,

where (hundreds of) Internet routers are connected. Because of the requirement to connect all routers to a single layer-2 network the peering networks use IPv4 layer-3 addresses in length ranges from /21 to /24, which can create very large broadcast domains. This peering network is transparent to the Customer Edge (CE) devices and therefore floods any ARP request or NS messages to all the CEs in the network. Unsolicited GARP and NA messages are flooded to all the CEs too.

In these IXP networks, most of the CEs are typically peering routers and roughly all the BUM traffic is originated by the ARP and ND address resolution procedures. This ARP/ND BUM traffic causes significant data volumes that reach every single router in the peering network. Since the ARP/ND messages are processed in software processors and they take high priority in the routers, heavy loads of ARP/ND traffic can cause some routers to run out of resources. CEs disappearing from the network may cause Address Resolution explosions that can make a router with limited processing power fail to keep BGP sessions running.

The issue may be better in IPv6 routers, since ND uses SN-multicast address in NS messages, however ARP uses broadcast and has to be processed by all the routers in the network. Some routers may also be configured to broadcast periodic GARPs [RFC5227]. The amount of ARP/ND flooded traffic grows exponentially with the number of IXP participants, therefore the issue can only go worse as new CEs are added.

In order to deal with this issue, IXPs have developed certain solutions over the past years. One example is the ARP-Sponge daemon [ARP-Sponge]. While these solutions may mitigate the issues of Address Resolution in large broadcasts domains, EVPN provides new more efficient possibilities to IXPs. EVPN and its proxy-ARP/ND function may help solve the issue in a distributed and scalable way, fully integrated with the PE network.

3. Solution Requirements

The distributed EVPN proxy-ARP/ND function described in this document SHOULD meet the following requirements:

- The solution SHOULD support the learning of the CE IP->MAC entries on the EVPN PEs via the management, control or data planes. An implementation SHOULD allow to intentionally enable or disable those possible learning mechanisms.

- The solution MAY suppress completely the flooding of the ARP/ND messages in the EVPN network, assuming that all the CE IP->MAC

addresses local to the PEs are known or provisioned on the PEs from a management system. Note that in this case, the unknown unicast traffic can also be suppressed, since all the expected unicast traffic will be destined to known MAC addresses in the PE MAC-VRFs.

- The solution MAY reduce significantly the flooding of the ARP/ND messages in the EVPN network, assuming that some or all the CE IP->MAC addresses are learned on the data plane by snooping ARP/ND messages issued by the CEs.

- The solution MAY provide a way to refresh periodically the CE IP->MAC entries learned through the data plane, so that the IP->MAC entries are not withdrawn by EVPN when they age out unless the CE is not active anymore. This option helps reducing the EVPN control plane overhead in a network with active CEs that do not send packets frequently.

- The solution SHOULD provide a mechanism to detect duplicate IP addresses. In case of duplication, the detecting PE should not reply to requests for the duplicate IP. Instead, the PE should alert the operator and may optionally prevent any other CE from sending traffic to the duplicate IP.

- The solution MUST NOT change any existing behavior in the CEs connected to the EVPN PEs.

4. Solution Description

Figure 1 illustrates an example EVPN network where the Proxy-ARP/ND function is enabled.

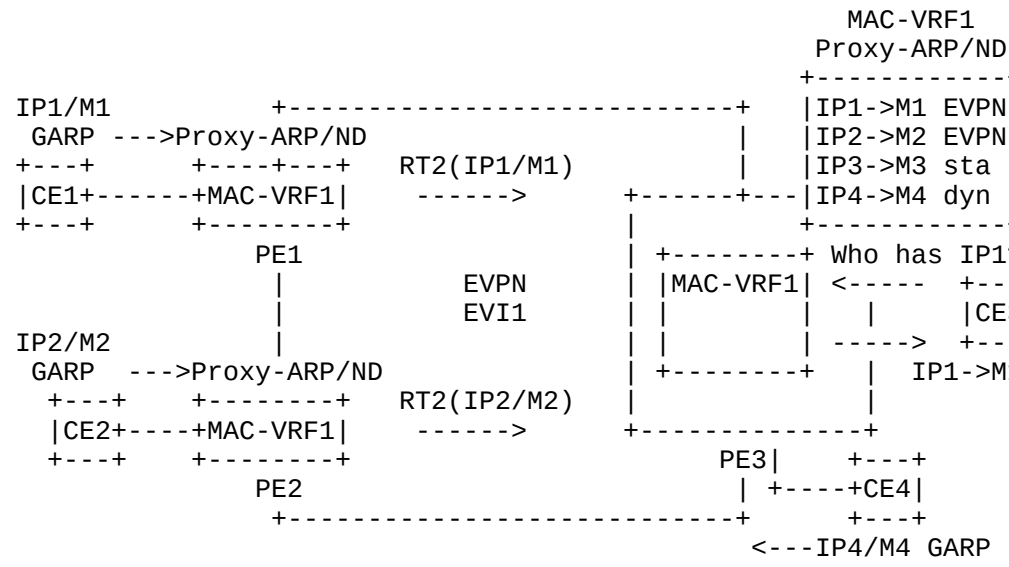


Figure 1 Proxy-ARP/ND network example

When the Proxy-ARP/ND function is enabled in the MAC-VRFs of the EVPN PEs, each PE creates a Proxy table specific to that MAC-VRF that can contain three types of Proxy-ARP/ND entries:

a) Dynamic entries: learned by snooping CE's ARP and ND messages. For instance, IP4->M4 in Figure 1.

b) Static entries: provisioned on the PE by the management system. For instance, IP3->M3 in Figure 1.

c) EVPN-learned entries: learned from the IP/MAC information encoded in the received RT2's coming from remote PEs. For instance, IP1->M1 and IP2->M2 in Figure 1.

As a high level example, the operation of the EVPN Proxy-ARP/ND function in the network of Figure 1 is described below. In this example we assume IP1, IP2 and IP3 are IPv4 addresses:

1. Proxy-ARP/ND is enabled in MAC-VRF1 of PE1, PE2 and PE3.

2. The PEs start adding dynamic, static and EVPN-learned entries to their Proxy tables:

   a. PE3 adds IP1->M1 and IP2->M2 based on the EVPN routes received from PE1 and PE2. Those entries were previously learned as dynamic entries in PE1 and PE2 respectively, and advertised in

BGP EVPN.
b. PE3 adds IP4->M4 as dynamic. This entry is learned by snooping the corresponding ARP messages sent by CE4.
c. An operator also provisions the static entry IP3->M3.

3. When CE3 sends an ARP Request asking for IP1, PE3 will:

a. Intercept the ARP Request and perform a Proxy-ARP lookup for IP1.
b. If the lookup is successful (as in Figure 1), PE3 will send an ARP Reply with IP1->M1. The ARP Request will not be flooded to the EVPN network or any other local CEs.
c. If the lookup is not successful, PE3 will flood the ARP Request in the EVPN network and the other local CEs.

As PE3 learns more and more host entries in the Proxy-ARP/ND table, the flooding of ARP Request messages is reduced and in some cases it can even be suppressed. In a network where most of the participant CEs are not moving between PEs and they advertise their presence with GARPs or unsolicited NA messages, the ARP/ND flooding as well as the unknown unicast flooding can practically be suppressed. In an EVPN-based IXP network, where all the entries are Static, the ARP/ND flooding is in fact totally suppressed.

The Proxy-ARP/ND function can be structured in five sub-functions or procedures:

1. Learning sub-function
2. Reply sub-function
3. Maintenance sub-function
4. Flooding reduction/suppression sub-function
5. Duplicate IP detection sub-function

A Proxy-ARP/ND implementation MAY support all those sub-functions or only a subset of them. The following sections describe each individual sub-function.

4.1. Learning Sub-Function

A Proxy-ARP/ND implementation SHOULD support static, dynamic and EVPN-learned entries.

Static entries are provisioned from the management plane. The provisioned static IP->MAC entry SHOULD be advertised in EVPN with a MAC Mobility extended community where the static flag is set to 1, as per [RFC7432]. A static entry MAY associate and IP to a list of potential MACs, i.e. IP1->(MAC1,MAC2..MACN). When there is more than

one MAC in the list of allowed MACs, the PE will not advertise any IP->MAC in EVPN until a local ARP/NA message or any other frame is received from the CE. Upon receiving traffic from the CE, the PE will check that the source MAC is included in the list of allowed MACs. Only in that case, the PE will activate the IP->MAC and advertise it in EVPN.

EVPN-learned entries MUST be learned from received valid EVPN MAC/IP Advertisement routes containing a MAC and IP address.

Dynamic entries are learned in different ways depending on whether the entry contains an IPv4 or IPv6 address:

a) Proxy-ARP dynamic entries:

   They SHOULD be learned by snooping any ARP packet (Ethertype 0x0806) received from the CEs attached to the MAC-VRF. The Learning function will add the Sender MAC and Sender IP of the snooped ARP packet to the Proxy-ARP table.

b) Proxy-ND dynamic entries:

   They SHOULD be learned out of the Target Address and TLLA information in NA messages (Ethertype 0x86DD, ICMPv6 type 136) received from the CEs attached to the MAC-VRF. A Proxy-ND implementation SHOULD NOT learn IP->MAC entries from NS messages, since they don’t contain the R-bit Flag required by the Proxy-ND reply function. See section 4.1.1 for more information about the R-bit flag.

   Note that if the O-bit is zero in the received NA message, the IP->MAC SHOULD only be learned in case IPv6 ’anycast’ is enabled in the EVI.

The following procedure associated to the Learning sub-function is recommended:

o When a new Proxy-ARP/ND EVPN or static active entry is learned (or provisioned), the PE SHOULD send an unsolicited GARP or NA message to the access CEs. The PE SHOULD send an unsolicited GARP/NA message for dynamic entries only if the ARP/NA message creating the entry was NOT flooded before. This unsolicited GARP/NA message makes sure the CE ARP/ND caches are updated even if the ARP/NS/NA messages from remote CEs are not flooded in the EVPN network.

Note that if a Static entry is provisioned with the same IP as an existing EVPN-learned or Dynamic entry, the Static entry takes precedence.

4.1.1. Proxy-ND and the NA Flags

[RFC4861] describes the use of the R-bit flag in IPv6 Address Resolution:

- Nodes capable of routing IPv6 packets must reply to NS messages with NA messages where the R-bit flag is set (R-bit=1).
- Hosts that are not able to route IPv6 packets must indicate that inability by replying with NA messages that contain R-bit=0.

The use of the R-bit flag in NA messages has an impact on how hosts select their default gateways when sending packets off-link:

- Hosts build a Default Router List based on the received RAs and NAs with R-bit=1. Each cache entry has an IsRouter flag, which must be set based on the R-bit flag in the received NAs. A host can choose one or more Default Routers when sending packets off-link.
- In those cases where the IsRouter flag changes from TRUE to FALSE as a result of a NA update, the node MUST remove that router from the Default Router List and update the Destination Cache entries for all destinations using that neighbor as a router, as specified in [RFC4861] section 7.3.3. This is needed to detect when a node that is used as a router stops forwarding packets due to being configured as a host.

The R-bit and O-bit will be learned in the following ways:

- Static entries SHOULD have the R-bit information added by the management interface. The O-bit information MAY also be added by the management interface.
- Dynamic entries SHOULD learn the R-bit and MAY learn the O-bit from the snooped NA messages used to learn the IP->MAC itself.
- EVPN-learned entries SHOULD learn the R-bit and MAY learn the O-bit from the ND Extended Community received from EVPN along with the RT2 used to learn the IP->MAC itself. Please refer to [EVPN-NA-FLAGS]. If no ND extended community is received, the PE will add the default R-bit/O-bit to the entry. The default R-bit SHOULD be an administrative choice. The default O-bit SHOULD be 1.

Note that the O-bit SHOULD only be learned if ’anycast’ is enabled in the EVI. If so, Duplicate IP Detection must be disabled so that the PE is able to learn the same IP mapped to different MACs in the same Proxy-ND table. If ’anycast’ is disabled, NA messages with O-bit = 0 will not create a proxy-ND entry, hence no EVPN advertisement with ND

extended community will be generated.

4.2. Reply Sub-Function

This sub-function will reply to Address Resolution requests/solicitations upon successful lookup in the Proxy-ARP/ND table for a given IP address. The following considerations should be taken into account:

a) When replying to ARP Request or NS messages, the PE SHOULD use the Proxy-ARP/ND entry MAC address as MAC SA. This is recommended so that the resolved MAC can be learned in the MAC FIB of potential Layer-2 switches seating between the PE and the CE requesting the Address Resolution.

b) A PE SHOULD NOT reply to a request/solicitation received on the same attachment circuit over which the IP->MAC is learned. In this case the requester and the requested IP are assumed to be connected to the same layer-2 switch/access network linked to the PE's attachment circuit, and therefore the requested IP owner will receive the request directly.

c) A PE SHOULD reply to broadcast/multicast Address Resolution messages, that is, ARP-Request, NS messages as well as DAD NS messages. A PE SHOULD NOT reply to unicast Address Resolution requests (for instance, NUD NS messages).

d) A PE SHOULD include the R-bit learned for the IP->MAC entry in the NA messages (see section 4.1.1). The S-bit will be set/unset as per [RFC4861]. The O-bit will be included if IPv6 'anycast' is enabled in the EVI and it is learned for the IP->MAC entry. If 'anycast' is enabled and there are more than one MAC for a given IP, the PE will reply to NS messages with as many NA responses as 'anycast' entries are in the proxy-ND table.

e) A PE SHOULD only reply to ARP-Request and NS messages with the format specified in [RFC0826] and [RFC4861] respectively. Received ARP-Requests and NS messages with unknown options SHOULD be either forwarded (as unicast packets) to the owner of the requested IP (assuming the MAC is known in the proxy-ARP/ND table and MAC-VRF) or discarded. An administrative option SHOULD control whether to 'unicast-forward' or 'discard' these frames with unknown options. Note that, as an example, this would allow to enable proxy-ND and Secure ND [RFC3971] in the same EVI. The 'unicast-forward' option allows the support of new unknown options in the EVI while reducing the flooding at the same time.

4.3. Maintenance Sub-Function

The Proxy-ARP/ND tables SHOULD follow a number of maintenance procedures so that the dynamic IP->MAC entries are kept if the owner is active and flushed if the owner is no longer in the network. The following procedures are recommended:

a) Age-time

A dynamic Proxy-ARP/ND entry SHOULD be flushed out of the table if the IP->MAC has not been refreshed within a given age-time. The entry is refreshed if an ARP or NA message is received for the same IP->MAC entry. The age-time is an administrative option and its value should be carefully chosen depending on the specific use-case: in IXP networks (where the CE routers are fairly static) the age-time may normally be longer than in DC networks (where mobility is required).

b) Send-refresh option

The PE MAY send periodic refresh messages (ARP/ND "probes") to the owners of the dynamic Proxy-ARP/ND entries, so that the entries can be refreshed before they age out. The owner of the IP->MAC entry would reply to the ARP/ND probe and the corresponding entry age-time reset. The periodic send-refresh timer is an administrative option and is recommended to be a third of the age-time or a half of the age-time in scaled networks.

An ARP refresh issued by the PE will be an ARP-Request message with the Sender’s IP = 0 sent from the PE’s MAC SA. An ND refresh will be a NS message issued from the PE’s MAC SA and a Link Local Address associated to the PE’s MAC.

The refresh request messages should be sent only for dynamic entries and not for static or EVPN-learned entries. Even though the refresh request messages are broadcast or multicast, the PE SHOULD only send the message to the attachment circuit associated to the MAC in the IP->MAC entry.

The age-time and send-refresh options are used in EVPN networks to avoid unnecessary EVPN RT2 withdrawals: if refresh messages are sent before the corresponding MAC-VRF FIB and Proxy-ARP/ND age-time for a given entry expires, inactive but existing hosts will reply, refreshing the entry and therefore avoiding unnecessary MAC and MAC-IP withdrawals in EVPN. Both entries (MAC in the MAC-VRF and IP->MAC in Proxy-ARP/ND) are reset when the owner replies to the ARP/ND probe. If there is no response to the ARP/ND probe, the MAC and IP->MAC entries will be legitimately flushed and the RT2s withdrawn.

4.4. Flooding (to Remote PEs) Reduction/Suppression

The Proxy-ARP/ND function implicitly helps reducing the flooding of ARP Request and NS messages to remote PEs in an EVPN network. However, in certain use-cases, the flooding of ARP/NS/NA messages (and even the unknown unicast flooding) to remote PEs can be suppressed completely in an EVPN network.

For instance, in an IXP network, since all the participant CEs are well known and will not move to a different PE, the IP->MAC entries may be all provisioned by a management system. Assuming the entries for the CEs are all provisioned on the local PE, a given Proxy-ARP/ND table will only contain static and EVPN-learned entries. In this case, the operator may choose to suppress the flooding of ARP/NS/NA to remote PEs completely.

The flooding may also be suppressed completely in IXP networks with dynamic Proxy-ARP/ND entries assuming that all the CEs are directly connected to the PEs and they all advertise their presence with a GARP/unsolicited-NA when they connect to the network.

In networks where fast mobility is expected (DC use-case), it is not recommended to suppress the flooding of unknown ARP-Requests/NS or GARPs/unsolicited-NAs. Unknown ARP-Requests/NS refer to those ARP-Request/NS messages for which the Proxy-ARP/ND lookups for the requested IPs do not succeed.

In order to give the operator the choice to suppress/allow the flooding to remote PEs, a PE MAY support administrative options to individually suppress/allow the flooding of:

o Unknown ARP-Request and NS messages.
o GARP and unsolicited-NA messages.

The operator will use these options based on the expected behavior in the CEs.

4.5. Duplicate IP Detection

The Proxy-ARP/ND function SHOULD support duplicate IP detection so that ARP/ND-spoofing attacks or duplicate IPs due to human errors can be detected.

ARP/ND spoofing is a technique whereby an attacker sends "fake" ARP/ND messages onto a broadcast domain. Generally the aim is to associate the attacker’s MAC address with the IP address of another host causing any traffic meant for that IP address to be sent to the

attacker instead.

The distributed nature of EVPN and proxy-ARP/ND allows the easy detection of duplicated IPs in the network, in a similar way to the MAC duplication function supported by [RFC7432] for MAC addresses.

Duplicate IP detection monitors "IP-moves" in the Proxy-ARP/ND table in the following way:

- When an existing active IP1->MAC1 entry is modified, a PE starts an M-second timer (default value of M=180), and if it detects N IP moves before the timer expires (default value of N=5), it concludes that a duplicate IP situation has occurred. An IP move is considered when, for instance, IP1->MAC1 is replaced by IP1->MAC2 in the Proxy-ARP/ND table.

- In order to detect the duplicate IP faster, the PE MAY send a CONFIRM message to the former owner of the IP. A CONFIRM message is a unicast ARP-Request/NS message sent by the PE to the MAC addresses that previously owned the IP, when the MAC changes in the Proxy-ARP/ND table. If the PE does not receive an answer within a given timer, the new entry will be confirmed and activated. For instance, if IP1->MAC1 moves to IP1->MAC2, the PE may send a unicast ARP-Request/NS message for IP1 with MAC DA= MAC1 and MAC SA= PE’s MAC. This will force the legitimate owner and the spoofer to reply so that the PE can detect the duplicate IP within the M timer:

  - If the IP1->MAC1 pair was previously owned by the spoofer and the new IP1->MAC2 was from a valid CE, then the issued CONFIRM message would trigger a response from the spoofer.

  - If it were the other way around, that is, IP1->MAC1 was previously owned by a valid CE, the CONFIRM message would trigger a response from the CE.

    Either way, if this process continues, then duplicate detection will kick in.

- Upon detecting a duplicate IP situation:

  a) The entry in duplicate detected state cannot be updated with new dynamic or EVPN-learned entries for the same IP. The operator MAY override the entry though with a static IP->MAC.

  b) The PE SHOULD alert the operator and stop responding ARP/NS for the duplicate IP until a corrective action is taken.

c) Optionally the PE MAY associate an "anti-spoofing-mac" (AS-MAC) to the duplicate IP. The PE will send a GARP/unsolicited-NA message with IP1->AS-MAC to the local CEs as well as an RT2 (with IP1->AS-MAC) to the remote PEs. This will force all the CEs in the EVI to use the AS-MAC as MAC DA for IP1, and prevent the spoofer from attracting any traffic for IP1. Since the AS-MAC is a managed MAC address known by all the PEs in the EVI, all the PEs MAY apply filters to drop and/or log any frame with MAC DA= AS-MAC. The advertisement of the AS-MAC as a "black-hole MAC" that can be used directly in the MAC-VRF to drop frames is for further study.

o The duplicate IP situation will be cleared when a corrective action is taken by the operator, or alternatively after a HOLD-DOWN timer (default value of 540 seconds).

The values of M, N and HOLD-DOWN timer SHOULD be a configurable administrative option to allow for the required flexibility in different scenarios.

For Proxy-ND, Duplicate IP Detection SHOULD only monitor IP moves for IP->MACs learned from NA messages with O-bit=1. NA messages with O-bit=0 would not override the ND cache entries for an existing IP. Duplicate IP Detection for IPv6 SHOULD be disabled when IPv6 ’anycast’ is activated in a given EVI.

5. Solution Benefits

The solution described in this document provides the following benefits:

a) The solution may suppress completely the flooding of the ARP/ND and unknown-unicast messages in the EVPN network, in cases where all the CE IP->MAC addresses local to the PEs are known and provisioned on the PEs from a management system.

b) The solution reduces significantly the flooding of the ARP/ND messages in the EVPN network, in cases where some or all the CE IP->MAC addresses are learned on the data plane by snooping ARP/ND messages issued by the CEs.

c) The solution reduces the control plane overhead and unnecessary BGP MAC/IP Advertisements and Withdrawals in a network with active CEs that do not send packets frequently.

d) The solution provides a mechanism to detect duplicate IP addresses and avoid ARP/ND-spoof attacks or the effects of duplicate

addresses due to human errors.

6. Deployment Scenarios

Four deployment scenarios with different levels of ARP/ND control are available to operators using this solution, depending on their requirements to manage ARP/ND: all dynamic learning, all dynamic learning with proxy-ARP/ND, hybrid dynamic learning and static provisioning with proxy-ARP/ND, and all static provisioning with proxy-ARP/ND.

6.1. All Dynamic Learning

In this scenario for minimum security and mitigation, EVPN is deployed in the peering network with the proxy-ARP/ND function shutdown. PEs do not intercept ARP/ND requests and flood all requests, as in a conventional layer-2 network. While no ARP/ND mitigation is used in this scenario, the IXP can still take advantage of EVPN features such as control plane learning and all-active multihoming in the peering network. Existing mitigation solutions, such as the ARP-Sponge daemon [ARP-Sponge] MAY also be used in this scenario.

Although this option does not require any of the procedures described in this document, it is added as baseline/default option for completeness.

6.2. Dynamic Learning with Proxy-ARP/ND

This scenario minimizes flooding while enabling dynamic learning of IP->MAC entries. The Proxy-ARP/ND function is enabled in the MAC-VRFs of the EVPN PEs, so that the PEs intercept and respond to CE requests.

The solution MAY further reduce the flooding of the ARP/ND messages in the EVPN network by snooping ARP/ND messages issued by the CEs.

PEs will flood requests if the entry is not in their Proxy table. Any unknown source MAC->IP entries will be learnt and advertised in EVPN, and traffic to unknown entries is discarded at the ingress PE.

6.3. Hybrid Dynamic Learning and Static Provisioning with Proxy-ARP/ND

Some IXPs want to protect particular hosts on the peering network while allowing dynamic learning of peering router addresses. For example, an IXP may want to configure static MAC->IP entries for management and infrastructure hosts that provide critical services. In this scenario, static entries are provisioned from the management

plane for protected MAC->IP addresses, and dynamic learning with Proxy-ARP/ND is enabled as described in section 6.2 on the peering network.

6.4 All Static Provisioning with Proxy-ARP/ND

For a solution that maximizes security and eliminates flooding and unknown unicast in the peering network, all MAC-IP entries are provisioned from the management plane. The Proxy-ARP/ND function is enabled in the MAC-VRFs of the EVPN PEs, so that the PEs intercept and respond to CE requests. Dynamic learning and ARP/ND snooping is disabled so that traffic to unknown entries is discarded at the ingress PE. This scenario provides and IXP the most control over MAC->IP entries and allows an IXP to manage all entries from a management system.

6.5 Deployment Scenarios in IXPs

Nowadays, almost all IXPs installed some security rules in order to protect the IXP-LAN. These rules are often called port security. Port security summarizes different operational steps that limit the access to the IXP-LAN, to the customer router and controls the kind of traffic that the routers are allowed to be exchange (e.g., Ethernet, IPv4, IPv6). Due to this, the deployment scenario as described in 6.4 "All Static Provisioning with Proxy-ARP/ND" is the predominant scenario for IXPs.

In addition to the "All Static Provisioning" behavior, in IXP networks it is recommended to configure the Reply Sub-Function to ’discard’ ARP-Requests/NS messages with unrecognized options.

At IXPs, customers usually follow a certain operational life-cycle. For each step of the operational life-cycle specific operational procedures are executed.

The following describes the operational procedures that are needed to guarantee port security throughout the life-cycle of a customer with focus on EVPN features:

1. A new customer is connected the first time to the IXP:

   Before the connection between the customer router and the IXP-LAN is activated, the MAC of the router is white-listed on the IXP’s switch port. All other MAC addresses are blocked. Pre-defined IPv4 and IPv6 addresses of the IXP’s peering network space are configured at the customer router. The IP->MAC static entries (IPv4 and IPv6) are configured in the management system of the IXP for the customer’s port in order to support Proxy-ARP/ND.

In case a customer uses multiple ports aggregated to a single logical port (LAG) some vendors randomly select the MAC address of the LAG from the different MAC addresses assigned to the ports. In this case the static entry will be used associated to a list of allowed MACs.

2. Replacement of customer router:

   If a customer router is about to be replaced, the new MAC address(es) must be installed in the management system besides the MAC address(es) of the currently connected router. This allows the customer to replace the router without any active involvement of the IXP operator. For this, static entries are also used. After the replacement takes place, the MAC address(es) of the replaced router can be removed.

3. Decommissioning a customer router

   If a customer router is decommissioned, the router is disconnected from the IXP PE. Right after that, the MAC address(es) of the router and IP->MAC bindings can be removed from the management system.

6.6 Deployment Scenarios in DCs

DCs normally have different requirements than IXPs in terms of Proxy-ARP/ND. Some differences are listed below:

a) The required mobility in virtualized DCs makes the "Dynamic Learning" or "Hybrid Dynamic and Static Provisioning" models more appropriate than the "All Static Provisioning" model.

b) IPv6 ’anycast’ may be required in DCs, while it is not a requirement in IXP networks. Therefore if the DC needs IPv6 ’anycast’ it will be explicitly enabled in the proxy-ND function, hence the proxy-ND sub-functions modified accordingly. For instance, if IPv6 ’anycast’ is enabled in the proxy-ND function, Duplicate IP Detection must be disabled.

c) DCs may require special options on ARP/ND as opposed to the Address Resolution function, which is the only one typically required in IXPs. Based on that, the Reply Sub-function may be modified to forward or discard unknown options.

7. Conventions Used in this Document

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

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

In this document, the characters ">>" preceding an indented line(s) indicates a compliance requirement statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the explicit compliance requirements of this RFC.

8. Security Considerations

When EVPN and its associated Proxy-ARP/ND function are used in IXP networks, they only provide ARP/ND security and mitigation. IXPs MUST still employ security mechanisms that protect the peering network and SHOULD follow established BCPs such as the ones described in [Euro-IX BCP].

For example, IXPs should disable all unneeded control protocols, and block unwanted protocols from CEs so that only IPv4, ARP and IPv6 Ethertypes are permitted on the peering network. In addition, port security features and ACLs can provide an additional level of security.

9. IANA Considerations

No IANA considerations.

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, <http://www.rfc-editor.org/info/rfc7432>.

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

[RFC0826]Plummer, D., "Ethernet Address Resolution Protocol: Or

Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, RFC 826, DOI 10.17487/RFC0826, November 1982, <http://www.rfc-editor.org/info/rfc826>.

[RFC6820]Narten, T., Karir, M., and I. Foo, "Address Resolution Problems in Large Data Center Networks", RFC 6820, DOI 10.17487/RFC6820, January 2013, <http://www.rfc-editor.org/info/rfc6820>.

[RFC7342]Dunbar, L., Kumari, W., and I. Gashinsky, "Practices for Scaling ARP and Neighbor Discovery (ND) in Large Data Centers", RFC 7342, DOI 10.17487/RFC7342, August 2014, <http://www.rfc-editor.org/info/rfc7342>.

[RFC3971]Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, DOI 10.17487/RFC3971, March 2005, <http://www.rfc-editor.org/info/rfc3971>.

[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>.

10.2. Informative References

[ARP-Sponge] Wessel M. and Sijm N., Universiteit van Amsterdam, "Effects of IPv4 and IPv6 address resolution on AMS-IX and the ARP Sponge", July 2009.

[EVPN-ND-FLAGS] Sathappan S., Nagaraj K. and Rabadan J., "Propagation of IPv6 Neighbor Advertisement Flags in EVPN", draft-snr-bess-evpn-na-flags-02, Work in Progress, July 2015.

[Euro-IX BCP] https://www.euro-ix.net/pages/28/1/bcp_ixp.html

11. Acknowledgments

The authors want to thank Ranganathan Boovaraghavan, Sriram Venkateswaran, Manish Krishnan, Seshagiri Venugopal, Tony Przygienda, Erik Nordmark and Robert Raszuk for their review and contributions. Thank you to Oliver Knapp as well, for his detailed review.

Authors' Addresses

Jorge Rabadan (Editor)
Alcatel-Lucent
777 E. Middlefield Road
Mountain View, CA 94043 USA
Email: jorge.rabadan@alcatel-lucent.com

Senthil Sathappan
Alcatel-Lucent
Email: senthil.sathappan@alcatel-lucent.com

Kiran Nagaraj
Alcatel-Lucent
Email: kiran.nagaraj@alcatel-lucent.com

Wim Henderickx
Alcatel-Lucent
Email: wim.henderickx@alcatel-lucent.com

Greg Hankins
Alcatel-Lucent
Email: greg.hankins@alcatel-lucent.com

Thomas King
DE-CIX
Email: thomas.king@de-cix.net

Daniel Melzer
DE-CIX
Email: daniel.melzer@de-cix.net

BESS
Internet-Draft
Updates: 7432 (if approved)
Intended status: Standards Track
Expires: October 23, 2016

Z. Zhang
W. Lin
Juniper Networks
J. Rabadan
Nokia
K. Patel
Cisco Systems

April 21, 2016

Updates on EVPN BUM Procedures
draft-zzhang-bess-evpn-bum-procedure-updates-03

Abstract

This document specifies procedure updates for broadcast, unknown unicast, and multicast (BUM) traffic in Ethernet VPNs (EVPN), including selective multicast, and provider tunnel segmentation.

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.

Status of This Memo

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

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

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

This Internet-Draft will expire on October 23, 2016.

Copyright Notice

Table of Contents

1. Terminology

To be added

2. Introduction

RFC 7432 specifies procedures to handle broadcast, unknown unicast, and multicast (BUM) traffic in Section 11, 12 and 16, using Inclusive Multicast Ethernet Tag Route. A lot of details are referred to RFC

7117 (VPLS Multicast). In particular, selective multicast is briefly mentioned for Ingress Replication but referred to RFC 7117.

RFC 7117 specifies procedures for using both inclusive tunnels and selective tunnels, similar to MVPN procedures specified in RFC 6513 and RFC 6514. A new SAFI "MCAST-VPLS" is introduced, with two types of NLRIs that match MVPN's S-PMSI A-D routes and Leaf A-D routes. The same procedures can be applied to EVPN selective multicast for both Ingress Replication and other tunnel types, but new route types need to be defined under the same EVPN SAFI.

MVPN uses terms I-PMSI and S-PMSI A-D Routes. For consistency and convenience, this document will use the same I/S-PMSI terms for VPLS and EVPN. In particular, EVPN's Inclusive Multicast Ethernet Tag Route and VPLS's VPLS A-D route carrying PTA (PMSI Tunnel Attribute) for BUM traffic purpose will all be referred to as I-PMSI A-D routes. Depending on the context, they may be used interchangeably.

MVPN provider tunnels and EVPN/VPLS BUM provider tunnels, which are referred to as MVPN/EVPN/VPLS provider tunnels in this document for simplicity, can be segmented for technical or administrative reasons, which are summarized in Section 2.1 of this document. RFC 6513/6514 cover MVPN inter-as segmentation, RFC 7117 covers VPLS multicast inter-as segmentation, and RFC 7524 (Seamless MPLS Multicast) covers inter-area segmentation for both MVPN and VPLS.

There is a difference between MVPN and VPLS multicast inter-as segmentation. For simplicity, EVPN uses the same procedures as in MVPN. All ASBRs can re-advertise their choice of the best route. Each can become the root of its intra-AS segment and inject traffic it receives from its upstream, while each downstream PE/ASBR will only pick one of the upstream ASBRs as its upstream. This is also the behavior even for VPLS in case of inter-area segmentation.

For inter-area segmentation, RFC 7524 requires the use of Inter-area P2MP Segmented Next-Hop Extended Community (S-NH-EC), and the setting of "Leaf Information Required" (LIR) flag in PTA in certain situations. Either of these could be optional in case of EVPN. Removing these requirements would make the segmentation procedures transparent to ingress and egress PEs.

RFC 7524 assumes that segmentation happens at area borders. However, it could be at "regional" borders, where a region could be a sub-area, or even an entire AS plus its external links (Section 6). That would allow for more flexible deployment scenarios (e.g. for single-area provider networks).

This document specifies/clarifies/redefines certain/additional EVPN BUM procedures, with a salient goal that they're better aligned among MVPN, EVPN and VPLS. For brevity, only changes/additions to relevant RFC 7117 and RFC 7524 procedures are specified, instead of repeating the entire procedures. Note that these are to be applied to EVPN only, even though sometimes they may sound to be updates to RFC 7117/7524.

2.1. Reasons for Tunnel Segmentation

Tunnel segmentation may be required and/or desired because of administrative and/or technical reasons.

For example, an MVPN/VPLS/EVPN network may span multiple providers and Inter-AS Option-B has to be used, in which the end-to-end provider tunnels have to be segmented at and stitched by the ASBRs. Different providers may use different tunnel technologies (e.g., provider A uses Ingress Replication, provider B uses RSVP-TE P2MP while provider C uses mLDP). Even if they use the same tunnel technology like RSVP-TE P2MP, it may be impractical to set up the tunnels across provider boundaries.

The same situations may apply between the ASes and/or areas of a single provider. For example, the backbone area may use RSVP-TE P2MP tunnels while non-backbone areas may use mLDP tunnels.

Segmentation can also be used to divide an AS/area to smaller regions, so that control plane state and/or forwarding plane state/ burden can be limited to that of individual regions. For example, instead of Ingress Replicating to 100 PEs in the entire AS, with inter-area segmentation [RFC 7524] a PE only needs to replicate to local PEs and ABRs. The ABRs will further replicate to their downstream PEs and ABRs. This not only reduces the forwarding plane burden, but also reduces the leaf tracking burden in the control plane.

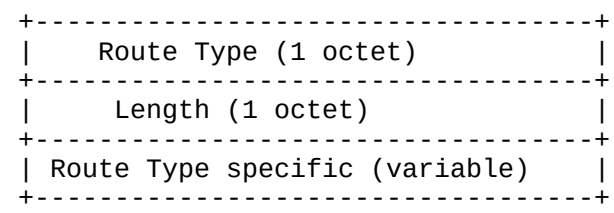
Smaller regions also have the benefit that, in case of tunnel aggregation, it is easier to find congruence among the segments of different constituent (service) tunnels and the resulting aggregation (base) tunnel in a region. This leads to better bandwidth efficiency, because the more congruent they are, the fewer leaves of the base tunnel need to discard traffic when a service tunnel's segment does not need to receive the traffic (yet it is receiving the traffic due to aggregation).

Another advantage of the smaller region is smaller BIER sub-domains. In this new multicast architecture BIER, packets carry a BitString, in which the bits correspond to edge routers that needs to receive

traffic. Smaller sub-domains means smaller BitStrings can be used without having to send multiple copies of the same packet.

3. Additional Route Types of EVPN NLRI

RFC 7432 defines the format of EVPN NLRI as the following:



So far five types have been defined:

+ 1 - Ethernet Auto-Discovery (A-D) route
+ 2 - MAC/IP Advertisement route
+ 3 - Inclusive Multicast Ethernet Tag route
+ 4 - Ethernet Segment route
+ 5 - IP Prefix Route

This document defines three additional route types:

+ 6 - Per-Region I-PMSI A-D route
+ 7 - S-PMSI A-D route
+ 8 - Leaf A-D route

The "Route Type specific" field of the type 6 and type 7 EVPN NLRIs starts with a type 1 RD, whose Administrative sub-field MUST match that of the RD in all the EVPN routes from the same advertising router for a given EVI, except the Leaf A-D route (Section 3.3).

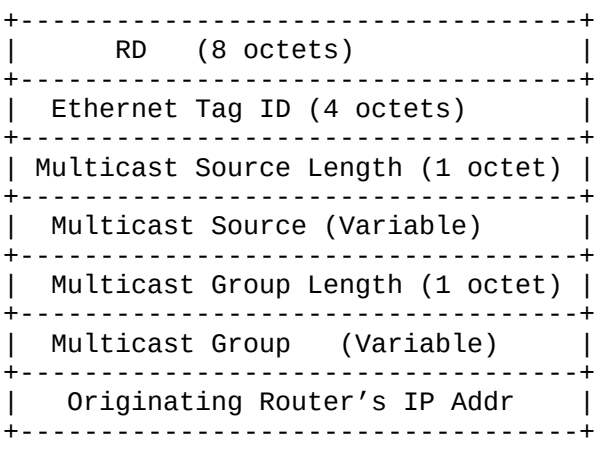
3.1. Per-Region I-PMSI A-D route

The Per-region I-PMSI A-D route has the following format. Its usage is discussed in Section 6.2.

After Ethernet Tag ID, an Extended Community (EC) is used to identify the region. Various types and sub-types of ECs provide maximum flexibility. Note that this is not an EC Attribute, but an 8-octet field embedded in the NLRI itself, following EC encoding scheme.

3.2. S-PMSI A-D route

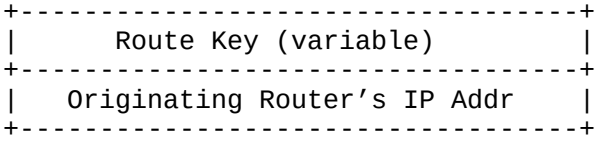
The S-PMSI A-D route has the following format:



Other than the addition of Ethernet Tag ID, it is identical to the S-PMSI A-D route as defined in RFC 7117. The procedures in RFC 7117 also apply (including wildcard functionality), except that the granularity level is per Ethernet Tag.

3.3. Leaf-AD route

The Route Type specific field of a Leaf A-D route consists of the following:



A Leaf A-D route is originated in response to a PMSI route, which could be an Inclusive Multicast Tag route, a per-region I-PMSI A-D route, an S-PMSI A-D route, or some other types of routes that may be defined in the future that triggers Leaf A-D routes. The Route Key is the "Route Type Specific" field of the route for which this Leaf A-D route is generated.

The general procedures of Leaf A-D route are first specified in RFC 6514 for MVPN. The principles apply to VPLS and EVPN as well. RFC 7117 has details for VPLS Multicast, and this document points out some specifics for EVPN, e.g. in Section 5.

4. Selective Multicast

RFC 7117 specifies Selective Multicast for VPLS. Other than that different route types and formats are specified with EVPN SAFI for S-PMSI A-D and Leaf A-D routes (Section 3), all procedures in RFC 7117 with respect to Selective Multicast apply to EVPN as well, including wildcard procedures.

5. Inter-AS Segmentation

5.1. Changes to Section 7.2.2 of RFC 7117

The first paragraph of Section 7.2.2.2 of RFC 7117 says:

> "... The best route procedures ensure that if multiple ASBRs, in an AS, receive the same Inter-AS A-D route from their EBGP neighbors, only one of these ASBRs propagates this route in Internal BGP (IBGP). This ASBR becomes the root of the intra-AS segment of the inter-AS tree and ensures that this is the only ASBR that accepts traffic into this AS from the inter-AS tree."

The above VPLS behavior requires complicated VPLS specific procedures for the ASBRs to reach agreement. For EVPN, a different approach is used and the above quoted text is not applicable to EVPN.

The Leaf A-D based procedure is used for each ASBR who re-advertises into the AS to discover the leaves on the segment rooted at itself. This is the same as the procedures for S-PMSI in RFC 7117 itself.

The following text at the end of the second bullet:

> "..... If, in order to instantiate the segment, the ASBR needs to know the leaves of the tree, then the ASBR obtains this information from the A-D routes received from other PEs/ASBRs in the ASBR’s own AS."

is changed to the following:

> "..... If, in order to instantiate the segment, the ASBR needs to know the leaves of the tree, then the ASBR MUST set the LIR flag to 1 in the PTA to trigger Leaf A-D routes from egress PEs and downstream ASBRs. It MUST be (auto-)configured with an import RT, which controls acceptance of leaf A-D routes by the ASBR."

Accordingly, the following paragraph in Section 7.2.2.4:

> "If the received Inter-AS A-D route carries the PMSI Tunnel attribute with the Tunnel Identifier set to RSVP-TE P2MP LSP, then the ASBR that originated the route MUST establish an RSVP-TE P2MP LSP with the local PE/ASBR as a leaf. This LSP MAY have been established before the local PE/ASBR receives the route, or it MAY be established after the local PE receives the route."

is changed to the following:

> "If the received Inter-AS A-D route has the LIR flag set in its PTA, then a receiving PE must originate a corresponding Leaf A-D route, and a receiving ASBR must originate a corresponding Leaf A-D route if and only if it received and imported one or more corresponding Leaf A-D routes from its downstream IBGP or EBGP peers, or it has non-null downstream forwarding state for the PIM/mLDP tunnel that instantiates its downstream intra-AS segment. The ASBR that (re-)advertised the Inter-AS A-D route then establishes a tunnel to the leaves discovered by the Leaf A-D routes."

5.2. I-PMSI Leaf Tracking

An ingress PE does not set the LIR flag in its I-PMSI's PTA, even with Ingress Replication or RSVP-TE P2MP tunnels. It does not rely on the Leaf A-D routes to discover leaves in its AS, and Section 11.2 of RFC 7432 explicitly states that the LIR flag must be set to zero.

An implementation of RFC 7432 might have used the Originating Router's IP Address field of the Inclusive Multicast Ethernet Tag routes to determine the leaves, or might have used the Next Hop field instead. Within the same AS, both will lead to the same result.

With segmentation, an ingress PE MUST determine the leaves in its AS from the BGP next hops in all its received I-PMSI A-D routes, so it does not have to set the LIR bit set to request Leaf A-D routes. PEs within the same AS will all have different next hops in their I-PMSI A-D routes (hence will all be considered as leaves), and PEs from other ASes will have the next hop in their I-PMSI A-D routes set to addresses of ASBRs in this local AS, hence only those ASBRs will be considered as leaves (as proxies for those PEs in other ASes). Note

that in case of Ingress Replication, when an ASBR re-advertises IBGP I-PMSI A-D routes, it MUST advertise the same label for all those for the same Ethernet Tag ID and the same EVI. When an ingress PE builds its flooding list, multiple routes may have the same (nexthop, label) tuple and they will only be added as a single branch in the flooding list.

5.3. Backward Compatibility

The above procedures assume that all PEs are upgraded to support the segmentation procedures:

- An ingress PE uses the Next Hop instead of Originating Router's IP Address to determine leaves for the I-PMSI tunnel.

- An egress PE sends Leaf A-D routes in response to I-PMSI routes, if the PTA has the LIR flag set (by the re-advertising ASBRs).

- In case of Ingress Replication, when an ingress PE builds its flooding list, multiple I-PMSI routes may have the same (nexthop, label) tuple and only a single branch for those will be added in the flooding list.

If a deployment has legacy PEs that does not support the above, then a legacy ingress PE would include all PEs (including those in remote ASes) as leaves of the inclusive tunnel and try to send traffic to them directly (no segmentation), which is either undesired or not possible; a legacy egress PE would not send Leaf A-D routes so the ASBRs would not know to send external traffic to them.

To address this backward compatibility problem, the following procedure can be used (see Section 6.2 for per-PE/AS/region I-PMSI A-D routes):

- An upgraded PE indicates in its per-PE I-PMSI A-D route that it supports the new procedures. Details will be provided in a future revision.

- All per-PE I-PMSI A-D routes are restricted to the local AS and not propagated to external peers.

- The ASBRs in an AS originate per-region I-PMSI A-D routes and advertise to their external peers to advertise tunnels used to carry traffic from the local AS to other ASes. Depending on the types of tunnels being used, the LIR flag in the PTA may be set, in which case the downstream ASBRs and upgraded PEs will send Leaf A-D routes to pull traffic from their upstream ASBRs. In a particular downstream AS, one of the ASBRs is elected, based on

the per-region I-PMSI A-D routes for a particular source AS, to send traffic from that source AS to legacy PEs in the downstream AS. The traffic arrives at the elected ASBR on the tunnel announced in the best per-region I-PMSI A-D route for the source AS, that the ASBR has selected of all those that it received over EBGP or IBGP sessions. Details of the election procedure will be provided in a future revision.

- In an ingress AS, if and only if an ASBR has active downstream receivers (PEs and ASBRs), which are learned either explicitly via Leaf AD routes or implicitly via PIM join or mLDP label mapping, the ASBR originates a per-PE I-PMSI A-D route (i.e., regular Inclusive Multicast Ethernet Tag route) into the local AS, and stitches incoming per-PE I-PMSI tunnels into its per-region I-PMSI tunnel. With this, it gets traffic from local PEs and send to other ASes via the tunnel announced in its per-region I-PMSI A-D route.

Note that, even if there is no backward compatibility issue, the above procedures have the benefit of keeping all per-PE I-PMSI A-D routes in their local ASes, greatly reducing the flooding of the routes and their corresponding Leaf A-D routes (when needed), and the number of inter-as tunnels.

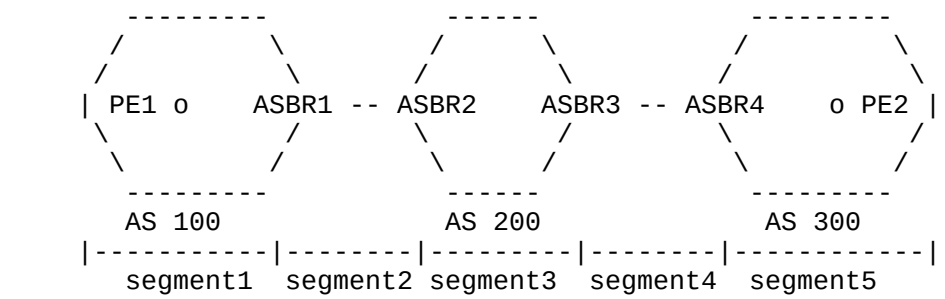
6. Inter-Region Segmentation

6.1. Area vs. Region

RFC 7524 is for MVPN/VPLS inter-area segmentation and does not explicitly cover EVPN. However, if "area" is replaced by "region" and "ABR" is replaced by "RBR" (Regional Border Router) then everything still works, and can be applied to EVPN as well.

A region can be a sub-area, or can be an entire AS including its external links. Instead of automatic region definition based on IGP areas, a region would be defined as a BGP peer group. In fact, even with IGP area based region definition, a BGP peer group listing the PEs and ABRs in an area is still needed.

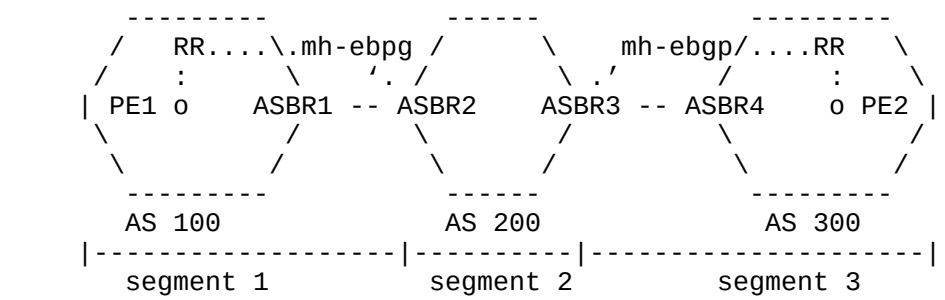
Consider the following example diagram:



The inter-as segmentation procedures specified so far (RFC 6513/6514, 7117, and Section 5 of this document) requires all ASBRs to be involved, and Ingress Replication is used between two ASBRs in different ASes.

In the above diagram, it's possible that ASBR1/4 does not support segmentation, and the provider tunnels in AS 100/300 can actually extend across the external link. In the case, the inter-region segmentation procedures can be used instead - a region is the entire (AS100 + ASBR1-ASBR2 link) or (AS300 + ASBR3-ASBR4 link). ASBR2/3 would be the RBRs, and ASBR1/4 will just be a transit core router with respect to provider tunnels.

As illustrated in the diagram below, ASBR2/3 will establish a multihop EBGP session with either a RR or directly with PEs in the neighboring AS. I/S-PMSI A-D routes from ingress PEs will not be processed by ASBR1/4. When ASBR2 re-advertises the routes into AS 200, it changes the next hop to its own address and changes PTA to specify the tunnel type/identification in its own AS. When ASBR3 re-advertises I/S-PMSI A-D routes into the neighboring AS 300, it changes the next hop to its own address and changes PTA to specify the tunnel type/identification in the neighboring region 3. Now the segment is rooted at ASBR3 and extends across the external link to PEs.



6.2. Per-region Aggregation

Notice that every I/S-PMSI route from each PE will be propagated throughout all the ASes or regions. They may also trigger corresponding Leaf A-D routes depending on the types of tunnels used in each region. This may become too many - routes and corresponding tunnels. To address this concern, the I-PMSI routes from all PEs in a AS/region can be aggregated into a single I-PMSI route originated from the RBRs, and traffic from all those individual I-PMSI tunnels will be switched into the single I-PMSI tunnel. This is like the MVPN Inter-AS I-PMSI route originated by ASBRs.

The MVPN Inter-AS I-PMSI A-D route can be better called as per-AS I-PMSI A-D route, to be compared against the (per-PE) Intra-AS I-PMSI A-D routes originated by each PE. In this document we will call it as per-region I-PMSI A-D route, in case we want to apply the aggregation at regional level. The per-PE I-PMSI routes will not be propagated to other regions. If multiple RBRs are connected to a region, then each will advertise such a route, with the same route key (Section 3.1). Similar to the per-PE I-PMSI A-D routes, RBRs/PEs in a downstream region will each select a best one from all those re-advertised by the upstream RBRs, hence will only receive traffic injected by one of them.

MVPN does not aggregate S-PMSI routes from all PEs in an AS like it does for I-PMSIs routes, because the number of PEs that will advertise S-PMSI routes for the same (s,g) or (*,g) is small. This is also the case for EVPN, i.e., there is no per-region S-PMSI routes.

Notice that per-region I-PMSI routes can also be used to address backwards compatibility issue, as discussed in Section 5.3.

The per-region I-PMSI route uses an embedded EC in NLRI to identify a region. As long as it uniquely identifies the region and the RBRs for the same region uses the same EC it is permitted. In the case where an AS number or area ID is needed, the following can be used:

- For a two-octet AS number, a Transitive Two-Octet AS-Specific EC of sub-type 0x09 (Source AS), with the Global Administrator sub-field set to the AS number and the Local Administrator sub-field set to 0.

- For a four-octet AS number, a Transitive Four-Octet AS-Specific EC of sub-type 0x09 (Source AS), with the Global Administrator sub-field set to the AS number and the Local Administrator sub-field set to 0.

- For an area ID, a Transitive IPv4-Address-Specific EC of any sub-type.

Uses of other particular ECs may be specified in other documents.

6.3. Use of S-NH-EC

RFC 7524 specifies the use of S-NH-EC because it does not allow ABRs to change the BGP next hop when they re-advertise I/S-PMSI AD routes to downstream areas. That is only to be consistent with the MVPN Inter-AS I-PMSI A-D routes, whose next hop must not be changed when they’re re-advertised by the segmenting ABRs for reasons specific to MVPN. For EVPN, it is perfectly fine to change the next hop when RBRs re-advertise the I/S-PMSI A-D routes, instead of relying on S-NH-EC. As a result, this document specifies that RBRs change the BGP next hop when they re-advertise I/S-PMSI A-D routes and do not use S-NH-EC. if a downstream PE/RBR needs to originate Leaf A-D routes, it simply uses the BGP next hop in the corresponding I/S-PMSI A-D routes to construct Route Targets.

The advantage of this is that neither ingress nor egress PEs need to understand/use S-NH-EC, and consistent procedure (based on BGP next hop) is used for both inter-as and inter-region segmentation.

6.4. Ingress PE’s I-PMSI Leaf Tracking

RFC 7524 specifies that when an ingress PE/ASBR (re-)advertises an VPLS I-PMSI A-D route, it sets the LIR flag to 1 in the route’s PTA. Similar to the inter-as case, this is actually not really needed for EVPN. To be consistent with the inter-as case, the ingress PE does not set the LIR flag in its originated I-PMSI A-D routes, and determines the leaves based on the BGP next hops in its received I-PMSI A-D routes, as specified in Section 5.2.

The same backward compatibility issue exists, and the same solution as in the inter-as case applies, as specified in Section 5.3.

7. Multi-homing Support

If multi-homing does not span across different ASes or regions, existing procedures work with segmentation. If an ES is multi-homed to PEs in different ASes or regions, additional procedures are needed to work with segmentation. The procedures are well understood but omitted here until the requirement becomes clear.

8. Security Considerations

This document does not seem to introduce new security risks, though this may be revised after further review and scrutiny.

9. Acknowledgements

The authors thank Eric Rosen, John Drake, and Ron Bonica for their comments and suggestions.

10. Contributors

The following also contributed to this document through their earlier work in EVPN selective multicast.

Junlin Zhang
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: jackey.zhang@huawei.com

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: lizhenbin@huawei.com

11. References

11.1. Normative References

[I-D.ietf-bess-ir] Rosen, E., Subramanian, K., and J. Zhang, "Ingress Replication Tunnels in Multicast VPN", draft-ietf-bess-ir-00 (work in progress), January 2015.

[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>.

[RFC7117] Aggarwal, R., Ed., Kamite, Y., Fang, L., Rekhter, Y., and C. Kodeboniya, "Multicast in Virtual Private LAN Service (VPLS)", RFC 7117, DOI 10.17487/RFC7117, February 2014, <http://www.rfc-editor.org/info/rfc7117>.

[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>.

[RFC7524] Rekhter, Y., Rosen, E., Aggarwal, R., Morin, T., Grosclaude, I., Leymann, N., and S. Saad, "Inter-Area Point-to-Multipoint (P2MP) Segmented Label Switched Paths (LSPs)", RFC 7524, DOI 10.17487/RFC7524, May 2015, <http://www.rfc-editor.org/info/rfc7524>.

11.2. Informative References

[I-D.ietf-bess-dci-evpn-overlay]
Rabadan, J., Sathappan, S., Henderickx, W., Palislamovic, S., Balus, F., Sajassi, A., and D. Cai, "Interconnect Solution for EVPN Overlay networks", draft-ietf-bess-dci-evpn-overlay-00 (work in progress), January 2015.

[I-D.ietf-bess-evpn-overlay]
Sajassi, A., Drake, J., Bitar, N., Isaac, A., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution using EVPN", draft-ietf-bess-evpn-overlay-01 (work in progress), February 2015.

[I-D.rabadan-bess-evpn-optimized-ir]
Rabadan, J., Sathappan, S., Henderickx, W., Sajassi, A., and A. Isaac, "Optimized Ingress Replication solution for EVPN", draft-rabadan-bess-evpn-optimized-ir-00 (work in progress), October 2014.

[I-D.wijnands-bier-architecture]
Wijnands, I., Rosen, E., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast using Bit Index Explicit Replication", draft-wijnands-bier-architecture-05 (work in progress), March 2015.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <http://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, <http://www.rfc-editor.org/info/rfc6514>.

Authors' Addresses

Zhaohui Zhang
Juniper Networks

EMail: zzhang@juniper.net

Wen Lin
Juniper Networks

EMail: wlin@juniper.net

Jorge Rabadan
Nokia

EMail: jorge.rabadan@nokia.com

Keyur Patel
Cisco Systems

EMail: keyupate@cisco.com