

INTERNET-DRAFT
Intended Status: Informational
Expires: January 4, 2015

Luyuan Fang
Microsoft
Rex Fernando
Dhananjaya Rao
Sami Boutros
Cisco

July 4, 2014

BGP/MPLS IP VPN Data Center Interconnect
draft-fang-l3vpn-data-center-interconnect-03

Abstract

This document discusses two categories of inter-connections of BGP/MPLS IP VPN and Data Center (DC) overlay networks. In the first category, DC overlay virtual network is built with BGP/MPLS IP VPN (IP VPN) technologies, and the inter-connection of IP VPN in the DC either to IP VPN in other DCs or to IP VPN in the WAN enables end-to-end IP VPN connectivity. In the second category, DC overlay network uses non IP VPN overlay technologies, and the inter-connection of any overlay virtual network in the DC to IP VPN in the WAN provides end user connectivity through stitching of different overlay technologies.

Status of this Memo

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

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

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

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

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

Copyright and License Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

1	Introduction	2
1.1	Terminology	3
2.	Use Cases	3
2.1	Case 1: End-to-end BGP IP VPN cloud inter-connection	4
2.2	Case 2: Hybrid cloud inter-connection	4
3.	Architecture reference models	4
3.1	BGP/MPLS IP VPN Inter-AS model	4
3.2	BGP/MPLS IP VPN Gateway PE to DC vCE Model	5
3.3	Hybrid inter-connection model	6
4.	Inter-connect IP VPN between DC and WAN	7
4.1	Existing Inter-AS options and DCI gap analysis	7
4.1.1	Option A pros and cons	7
4.1.2	Option B pros and cons	8
4.1.3	Option C pros and cons	8
4.1.4	Use of RTC	9
5.	Inter-connect IP VPN and non-IP VPN overlay networks	9
6.	Security Considerations	10
7.	IANA Considerations	10
8.	References	11
8.1	Normative References	11
8.2	Informative References	11
	Authors' Addresses	12

1 Introduction

With the growth of cloud services, the need of inter-connecting DC overlay networks and Enterprise BGP/MPLS IP VPNs in the Wide Area Network (WAN) arises.

Two categories of inter-connections of BGP/MPLS IP VPN [RFC4364] and Data Center (DC) overlay networks are discussed in this document. In the first category, DC overlay virtual network is built with BGP/MPLS IP VPN (IP VPN) technologies, and the inter-connection of IP VPN in the DC either to IP VPN in other DCs or to IP VPN in the WAN enables end-to-end IP VPN connectivity for Virtual Private Cloud (VPC) services. In the second category, DC overlay network uses non IP VPN overlay technologies, the inter-connection of any overlay virtual network in the DC to IP VPN in the WAN provides end user connectivity through stitching of different overlay technologies.

This document discusses use cases of the inter-connection of BGP/MPLS VPN to Data Centers, the general requirements, and the proposed solutions for the inter-connections.

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].

Term	Definition
-----	-----
AS	Autonomous System
ASBR	Autonomous System Border Router
BGP	Border Gateway Protocol
CE	Customer Edge
GRE	Generic Routing Encapsulation
Hypervisor	Virtual Machine Manager
I2RS	Interface to Routing System
MP-BGP	Multi-Protocol Border Gateway Protocol
NVGRE	Network Virtualization using GRE
QoS	Quality of Service
RD	Route Distinguisher
RR	Route Reflector
RT	Route Target
RTC	RT Constraint
SDN	Software Defined Network
ToR	Top-of-Rack switch
vCE	virtual Customer Edge Router
VM	Virtual Machine
VN	Virtual Network
VPC	Virtual Private Cloud
vPE	virtual Provider Edge Router
VPN	Virtual Private Network
VXLAN	Virtual eXtensible Local Area Network
WAN	Wide Area Network

2. Use Cases

2.1 Case 1: End-to-end BGP IP VPN cloud inter-connection

BGP/MPLS IP VPN is a proven scalable overlay technology with extensive deployment. It is an excellent candidate for end-to-end (host-to-host) overlay technology for Cloud-Scale DC application support. In addition, many SPs are interested to extend the IP VPN capabilities into their DCs to provide end-to-end native BGP IP VPN services to their enterprise customers.

BGP IP VPN provides routing isolation, rich policy control, and QoS support. The technologies developed to extend BGP IP VPN into DC servers or ToR are work in progress in IETF, [I-D.fang-l3vpn-virtual-pe], and [I-D.ietf-l3vpn-end-system].

The WAN and DC may be managed by the same or different administrative domains.

2.2 Case 2: Hybrid cloud inter-connection

Inter-connecting network SPs Enterprise IP VPNs to Cloud/Content providers DCs for expanded services. The inter-connection between the SP BGP/MPLS IP VPNs and the cloud provider networks is needed to provide the service end-to-end. The inter-connection of different types of providers can be BGP/MPLS IP VPN to other VPN or overlay technologies which may be virtualized or non-virtualized.

3. Architecture reference models

The architecture reference models described below focus on the inter-connection aspects. Although the intra-DC implementation is not within the scope of this discussion, the intra-DC technology has a direct impact to inter-DC connection. Therefore, various models are illustrated.

3.1 BGP/MPLS IP VPN Inter-AS model

The BGP/MPLS IP VPNs are implemented in both the WAN network and the Data Center. A customer VPN, for example VPNA in figure 1, consists of enterprise remote sites and VMs supporting applications in the DC. The IP VPN implementation is using vPE technology in DC. The two segments of the VPNs are inter-connected through ASBRs facing each other in the respective networks.

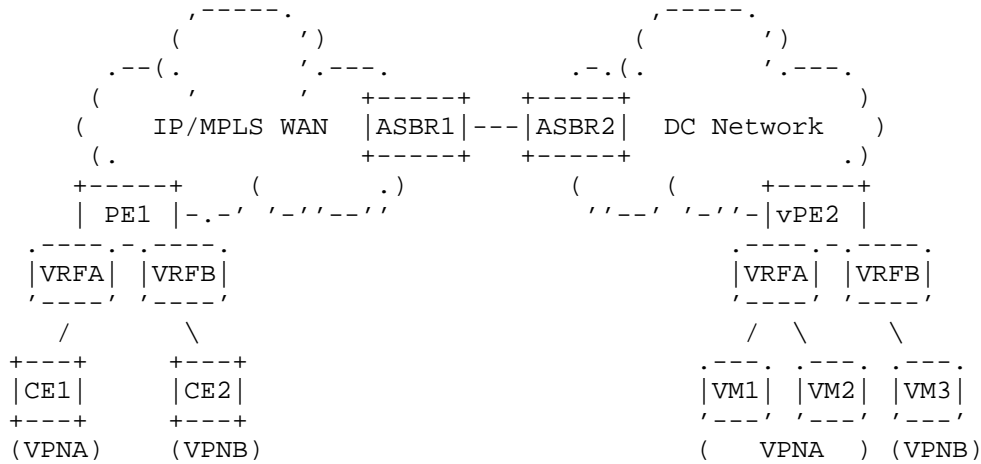


Figure 1. BGP/MPLS IP VPN Inter-Connection
with ASBR in each network

One boarding ASBR can be shared for the inter-connection of the two networks, especially if the WAN and DC belong to the same provider. Figure 2 illustrates this shared ASBR model.

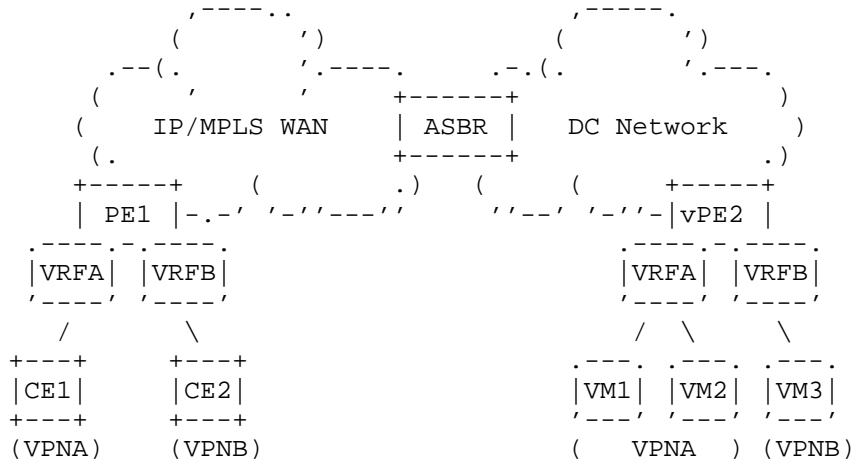


Figure 2. BGP/MPLS IP VPN Inter-Connection
with share ASBR

3.2 BGP/MPLS IP VPN Gateway PE to DC vCE Model

A simple virtual CE (vCE) [I-D.fang-l3vpn-virtual-ce] model can be used to inter-connect client containers to the DC Gateway which function as PE. This model is used by SPs to provide managed services, when scale can meet the service requirement.

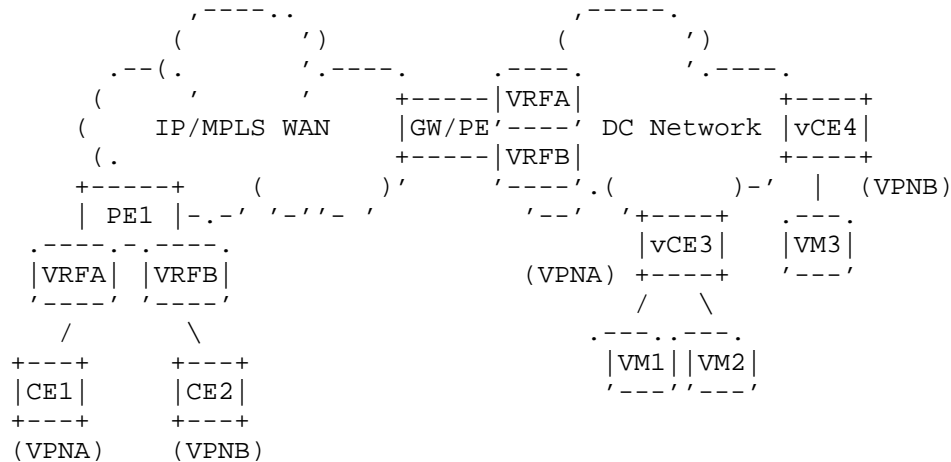


Figure 3. BGP/MPLS IP VPN GW/PE to vCEs
without BGP/MPLS IP VPN in the DC

3.3 Hybrid inter-connection model

The BGP/MPLS IP VPNs are implemented in the WAN network, and non BGP/MPLS IP VPN Overlay are implemented in the DC. The connection of the two networks is outside of the technologies for Inter-AS connections for BGP IP VPNs. This model includes many variations depending on the specific technologies used in the DC overlay. Figure 4 provides a general view of this inter-connecting model with ASBR on the MPLS WAN side, and the DC GW on the DC side. It is also viable to use one shared ASBR/GW for the inter-connection, especially if the WAN and the DC belong to the same provider.

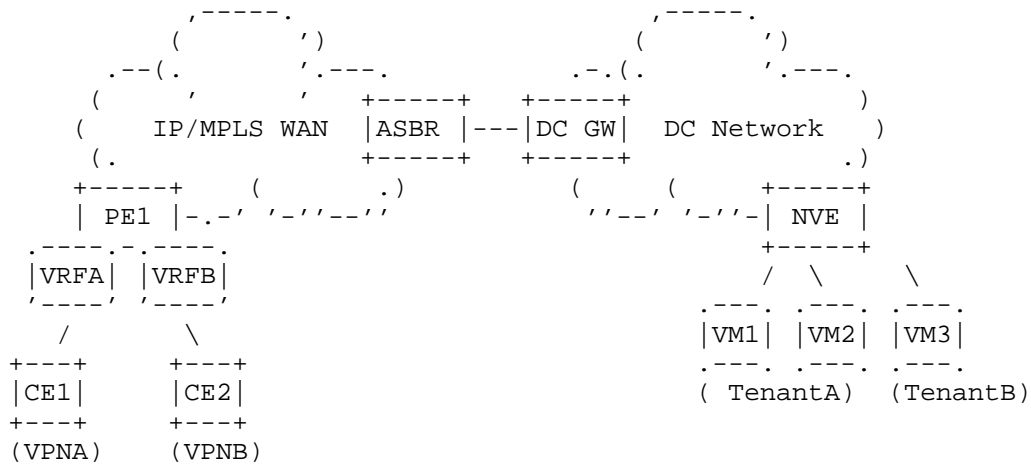


Figure 4. BGP/MPLS IP VPN Inter-Connection with
non BGP/MPLS IP VPN Overlay in DC

4. Inter-connect IP VPN between DC and WAN

4.1 Existing Inter-AS options and DCI gap analysis

The inter-AS options described in [RFC4364] can be used for DC inter-connection. Option A, B, and C must be supported.

4.1.1 Option A pros and cons

In Option A: back-to-back VRF. The PE-ASBR in one AS performs MPLS or IP VPN de-encapsulation and transmits packets to the peer PE-ASBR in the adjacent AS. The peer PE-ASBR performs MPLS or IP VPN encapsulation on the customer IPv4/IPv6 packets received, and transmits the packet through the IP backbone of the AS. VPN service providers exchange routes across a back-to-back VRF connection. Each VRF instance represents a separate VPN client, and it is configured on a separate PE-ASBR interface, allowing a PE-ASBR to communicate with its peer PE-ASBR as if the peer was a CE router.

Pros: This is the most secure option among options A, B, and C. And it is the simplest model from operation perspective. Each PE-ASBR is treating the other as a CE.

Cons: This option suffers from scaling limitations, because per Inter-AS VPN VRF and interface are needed on the PE-ASBR.

Option A has been commonly used in BGP/MPLS VPN Inter-Provider inter-

connections because of the security considerations and its clear operational demarcation.

DCI considerations: This is a simple way to connect DC and WAN if both sides are of small scale. Scale will be the major concern for DC inter-connect if large scale support is needed. Even if the DC scale is small, there are major concerns on receiving relevant routes (potentially a large number) from the WAN side, and Vice Versa.

4.1.2 Option B pros and cons

In Option B: EBGp redistribution of labeled VPN-IPv4/IPv6 routes between the neighboring ASes. ASes exchange VPN routing information (routes and labels) to establish connections. To control connections between ASes, the PE routers and EBGp border edge routers maintain a label forwarding information base (LFIB). The LFIB manages the labels and routes that the PE routers and EBGp border edge routers receive during the exchange of VPN information. The ASes exchange VPN routing information, such as, the destination network, the next hop field associated with the distributing router, a local MPLS label, and an RD. ASBRs are configured to change the next hop (next-hop-self) when sending VPN-IPv4 NLRI to the IBGP neighbors; the ASBRs must allocate a new label when they forward the NLRI to the IBGP neighbors.

Pros: It provides improved scalability when compared with option A, since it removes the needs of per Inter-AS VPN VRF and interface on the ASBR.

Cons: vanilla version of Option B is considered less secure in comparison with Option A, due to the dynamic routing information exchange that is involved. The ASBR scaling may still be an issue because ASBR must maintain all VPN routes.

Option B is commonly used within single provider or for inter-provider connections.

DCI considerations: Option B is one viable option to be used in DC inter-connection. However, it has the same scale concerns as other options because of the potentially large number of routes exchanged between the WAN and the DC.

4.1.3 Option C pros and cons

In option C: Multihop eBGp redistribution of labeled VPN-IPv4/IPv6 routes between source and destination ASs, with eBGp redistribution of labeled IPv4/IPv6 routes from AS to neighboring AS. The ASBRs need only to exchange host routes (/32 or /128) to the PE routers involved in the VPN, with the labels needed to get there. A Label Switch Path

(LSP) is built from the ingress PE router in one AS to the egress PE in the other AS (using Loopback addresses). VPN traffic uses this LSP to reach the other AS. From data plane's perspective, the ASBRs act as P routers, with no knowledge about the VPNs concerned. Between the two inter-connecting ASBRs, the VPN traffic is treated just as between two P routers, each VPN data packet is pre-pended with the VPN label and then with an egress-PE label. Option C can be further scaled by using route reflectors (RRs) in each AS.

Pros: It is the most scalable option among all three. ASBR is no longer a bottle neck for VPN routes scaling as in Option B.

Cons: Major security issues as IGP reachability must be exchanged between the inter-connecting ASes.

Option C has been used within a single SP for inter-AS connections. Using RR for VPN routes exchange is the common approach.

DCI consideration: Option C SHOULD NOT be used for any DCI which is between two different providers for security reasons.

In this option, though ASBR is no longer the scaling bottleneck, the scaling issues still call for careful design, as the total numbers of VRFs, VPN interfaces, and the VPN routes being exchanged between the two ASes can be very large.

4.1.4 Use of RTC

RT constraint [RFC4684] function must be used to only distribute the IP VPN routes of a VPN from one AS to another under the condition that they both support that VPN in each of the AS. This is one most important function for scalable solution.

However, all IP VPN routes are exchanged between the two ASes (e.g. WAN and DC) as long as they have to support the same VPNs. The potential IP VPN routes distribution can still be very substantial in large WAN and DC deployment. Additional aggregation and abstraction mechanisms can be used to avoid large numbers of VPN routes being exchanged across the border between the interconnecting WAN and the DC in either directions.

5. Inter-connect IP VPN and non-IP VPN overlay networks

As one significant instance of the hybrid use-case described in section 2.2, a DC may support a multi-tenant virtualized service network using IP based DC overlay encapsulations such as VXLAN [I-D.mahalingam-dutt-dcops-vxlan] or NVGRE [I-D.sridharan-virtualization-nvgre]. Different deployment models may

be used within the DC depending on the DC provider's functional and operational requirements.

When an IP DC overlay is terminated at the DC Gateway router and traffic directed into a BGP/MPLS IP VPN, the DC Gateway router performs MPLS encapsulation towards the WAN and IP overlay based forwarding within the DC.

The inter-connection mechanisms between the DC and the WAN may fall into two categories:

1. VRF Termination

The overlay based virtual network terminates into a BGP IP VPN VRF at the DC-WAN Gateway router. Both the internal routes of the DC as well as the external routes received from the WAN router can be installed in the VRF forwarding table at the DC Gateway router. The DC Gateway performs an IP lookup, appropriate MPLS or IP encapsulation, and forward traffic.

The DC Gateway router peers with the WAN router using one of the existing inter-AS mechanisms described above. The DC Gateway functions as an IP-VPN ASBR with local VRFs; for example, packets still undergo an IP forwarding lookup.

2. DC-VN and IP VPN Inter-working

In this case, the DC Gateway router performs a direct translation between VN-IDs and IP VPN labels while switching packets between the DC and WAN interfaces without performing an IP lookup. The forwarding table at the DC Gateway router is set up to do a VN-ID or label lookup and derive the output label or VN-ID. The DC Gateway Router acts as an Inter-AS Option B ASBR peering with other ASBRs.

6. Security Considerations

BGP/MPLS Inter-AS security threats and defense techniques described in RFC 4111 [RFC4111] are applicable for the WAN/DC inter-connections.

In addition, the protocols between the Gateway routers and the controller/orchestrator MUST be mutually authenticated. Given the potentially very large scale and the dynamic nature in the cloud/DC environment, the choice of key management mechanisms need to be further studied.

7. IANA Considerations

None.

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.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [RFC4684] Marques, P., Bonica, R., Fang, L., Martini, L., Raszuk, R., Patel, K., and J. Guichard, "Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)", RFC 4684, November 2006.

8.2 Informative References

- [RFC4111] Fang, L., Ed., "Security Framework for Provider-Provisioned Virtual Private Networks (PPVPNs)", RFC 4111, July 2005.
- [I-D.ietf-l3vpn-end-system] Marques, P., Fang, L., Pan, P., Shukla, A., Napierala, M., "BGP-signaled end-system IP/VPNs", draft-ietf-l3vpn-end-system, work in progress.
- [I-D.fang-l3vpn-virtual-pe] Fang, L., Ward, D., Fernando, R., Napierala, M., Bitar, N., Rao, D., Rijsman, B., So, N., "BGP IP VPN Virtual PE", draft-fang-l3vpn-virtual-pe, work in progress.
- [I-D.fang-l3vpn-virtual-ce] Fang, L., Evans, J., Ward, D., Fernando, R., Mullooly, J., So, N., Bitar, N., Napierala, M., "BGP IP VPN Virtual PE", draft-fang-l3vpn-virtual-ce, work in progress.
- [I-D.mahalingam-dutt-dcops-vxlan]: Mahalingam, M, Dutt, D., et al., "A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks" draft-mahalingam-dutt-dcops-vxlan, work in progress.
- [I-D.sridharan-virtualization-nvgre]: SridharanNetwork, M., et al., "Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre, work in progress.

Authors' Addresses

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

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

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

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

INTERNET-DRAFT
Intended Status: Standards track
Expires: January 4, 2015

Luyuan Fang
Microsoft
John Evans
David Ward
Rex Fernando
Cisco
Ning So
Vinci Systems
Nabil Bitar
Verizon
Maria Napierala
AT&T

July 4, 2014

BGP IP MPLS VPN Virtual CE
draft-fang-l3vpn-virtual-ce-03

Abstract

This document describes the architecture and solutions of using virtual Customer Edge (vCE) of BGP IP MPLS VPN. The solution is aimed at providing efficient service delivery capability through CE virtualization, and is especially beneficial in virtual Private Cloud (vPC) environments when extending IP MPLS VPN into tenant virtual Data Center containers. This document includes: BGP IP MPLS VPN virtual CE architecture; Control plane and forwarding options; Data Center orchestration processes; integration with existing WAN enterprise VPNs; management capability requirements; and security considerations. The solution is generally applicable to any BGP IP VPN deployment. The virtual CE solution is complementary to the virtual PE solutions.

Today's data center's require multi-tenancy and mechanisms to establish overlay network connectivity. This document describes one approach to enabling data center network connectivity.

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.

Expires <January 4, 2015>

[Page 1]

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

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

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

Copyright and License Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

1. Introduction	4
1.1 Terminology	4
1.2 Problem statement	6
1.3 Scope of the document	6
2. Virtual CE Architecture and Reference Model	7
2.1 Virtual CE	7
2.2 Architecture	8
3. Control Plane	10
3.1 vCE Control Plane	10
4. Forwarding Plane	11
4.1 Forwarding between vCE and PE/vPE	11
4.2 Forwarding between vCE and VM	11
5. Addressing and QoS	11
5.1 Addressing	11
5.2 QoS	12
6. Management plane	12
6.1 Network abstraction and management	12
6.2 Service VM Management	12

7. Orchestration and IP VPN inter-provisioning	12
7.1 DC Instance to WAN IP VPN instance "binding" Requirements	12
7.2. Provisioning/Orchestration	13
7.2.1 vCE Push model	13
7.2.1.1 Inter-domain provisioning vCE Push Model	14
7.2.1.2 Cross-domain provisioning vCE Push Model	14
7.2.1.1 vCE Pull model	15
8. vCE and vPE interaction	16
8.1 Traditional vCE-PE connectivity	16
8.2 vCE-vPE connectivity	16
8.2.1 Co-located vCE-vPE connectivity with vPE Model 1	17
8.2.2 Co-located vCE-vPE connectivity with vPE Model 2	18
8. Security Considerations	18
9. IANA Considerations	18
10. References	18
10.1 Normative References	18
10.2 Informative References	19
11. Acknowledgement	20
Authors' Addresses	20

1. Introduction

In the typical enterprise BGP/MPLS IP VPN [RFC4364] deployment, the Provider Edge (PE) and Customer Edge (CE) are physical routers which support the PE and CE functions. With the recent development of cloud services, using virtual instances of PE or CE functions, which reside in a compute device such as a server, can be beneficial to emulate the same logical functions as the physical deployment model but now achieved via cloud based network virtualization principles.

This document describes IP VPN virtual CE (vCE) solutions, while Virtual PE (vPE) concept and implementation options are discussed in [I-D.fang-l3vpn-virtual-pe], [I-D.ietf-l3vpn-end-system]. vPE and vCE solutions provide two avenues to realize network virtualization.

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].

Term	Definition
-----	-----
AAA	Authentication, Authorization, and Accounting
ACL	Access Control List
3GPP	3rd Generation Partnership Project (3GPP)
AS	Autonomous Systems
ASBR	Autonomous Systems Border Router
BFD	Bidirectional Forwarding Detection
BGP	Border Gateway Protocol
CE	Customer Edge
DB	Data Base
DMZ	Demilitarized Zone, a.k.a. perimeter networking
ED	End device: where Guest OS, Host OS/Hypervisor, applications, VMs, and virtual router may reside
FE	Front End
FIB	Forwarding Information Base
Forwarder	L3VPN forwarding function
FRR	Fast Re-Route
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HTTP	Hypertext Transfer Protocol
Hypervisor	Virtual Machine Manager
I2RS	Interface to Routing System
LDAP	Lightweight Directory Access Protocol
MP-BGP	Multi-Protocol Border Gateway Protocol

NVGRE	Network Virtualization using GRE
OSPF	Open Shortest Path First
PE	Provider Edge
QinQ	Provider Bridging, stacked VLANs
RR	Route Reflector
SDN	Software Defined Network
SLA	Service Level Agreement
SMTP	Simple Mail Transfer Protocol
ToR	Top of the Rack switch
VI	Virtual Interface
vCE	virtual Customer Edge Router
vLB	virtual Load Balancer
VM	Virtual Machine
VLAN	Virtual Local Area Network
vPC	virtual Private Cloud
vPE	virtual Provider Edge Router
VPN	Virtual Private Network
vRR	virtual Route Reflector
vSG	virtual Security Gateway
VXLAN	Virtual eXtensible Local Area Network
WAN	Wide Area Network

Definitions:

Virtual CE (vCE): A virtual instance of the Customer Edge (CE) routing function which resides in one or more network or compute devices. For example, the vCE data plane may reside in an end device, such as a server, and as co-resident with application Virtual Machines (VMs) on the server; the vCE control plane may reside in the same device or in a separate entity such as a controller.

Network Container/Tenant Container: An abstraction of a set of network and compute resources which can be physical and virtual, providing the cloud services for a tenant. One tenant can have more than one Tenant Containers.

Zone: A logical grouping of VMs and service assets within a tenant container. Different security policies may be applied within and between zones.

DMZ: Demilitarized zone, a.k.a. perimeter networking. It is often a machine or a small subnet that sits between a trusted internal network, such as a corporate private LAN, and an un-trusted external network, such as the public Internet. Typically, the DMZ contains devices accessible to Internet traffic, such as Web (HTTP) servers, FTP servers, SMTP (e-mail) servers and DNS servers.

1.2 Problem statement

With the growth of cloud services and the increase in the number of CE devices, routers/switches, and appliances, such as Firewalls (FWs) and Load Balancers (LBs), that need to be supported, there are benefits to virtualize the Data Center tenant container. The virtualized container can increase resource sharing, optimize routing and forwarding of inter-segment and inter-service traffic, and simplify design, provisioning, and management.

The following two aspects of the virtualized Data Center tenant container for the IP VPN CE solution are discussed in this document.

1. Architecture re-design for virtualized DC.

The optimal architecture of the virtualized container includes virtual CE, virtual appliances, application VMs. All these functions are co-residents on virtualized servers. In this arrangement, CEs and appliances can be created and removed easily on demand, and the virtual CE can interconnect the virtual appliances (e.g., FW, LB, NAT), applications (e.g., Web, App., and DB) in a co-located fashion for simplicity, routing/forwarding optimization, and easier service chaining. Virtualizing these functions on a per-tenant basis provides simplicity for the network operator in regards to managing per tenant service orchestration, tenant container moves, capacity planning across tenants and per-tenant policies.

2. Provisioning/orchestration. Two issues need to be addressed:

- a) The provisioning/orchestration system of the virtualized data center need to support VM life cycle and VM migration.
- b) The provisioning/orchestration systems of the DC and the WAN networks need to be coordinated to support end-to-end IP VPN from DC to DC or from DC to enterprise remote office in the same VPN. The DC and the WAN network are often operated by separate departments, even if they belong to the same provider. Today, the process of inter-connecting is slow and painful, and automation is highly desirable.

1.3 Scope of the document

It is assumed that the readers are familiar with BGP/MPLS IP VPN [RFC4364] terms and technologies, the base technology and its operation are not reviewed in details in this document.

As the majority (all in some networks) of applications are IP, this vCE solution is focusing on IP VPN solutions to cover the most common cases and keep matters as simple as possible.

2. Virtual CE Architecture and Reference Model

2.1 Virtual CE

As described in [RFC4364], IP uses a "peer model" - the customers' edge routers (CE routers) exchange routes with the Service Provider's edge routers (PE routers); the CEs do not peer with each other. MP-BGP [RFC4271, RFC4760] is used between the PEs (often with RRs) which have a particular VPN attached to them to exchange the VPN routes. A CE sends IP packets to the PE; no VPN labels for packets forwarded between CE and PE.

A virtual CE (vCE) as defined in this document is a software instance of IP VPN CE function which can reside in ANY network or compute devices. For example, a vCE MAY reside in an end device, such as a server in a Data Center, where the application VMs reside. The CE functionality and management models remain the same as defined in [RFC4364] regardless of whether the CE is physical or virtual.

Using the virtual CE model, the CE functions CAN easily co-located with the VM/applications, e.g., in the same server. This allows tenant inter-segment and inter-service routing to be optimized. Likewise the vCE can be in a separate server (in the same DC rack or across racks) than the application VMs, in which case VMs would typically use standard L2 technologies to access the vCE via the DC network.

Similar to the virtual PE solution, the control and forwarding of a virtual CE can be on the same device, or decoupled and reside on different physical devices. The provisioning of a virtual CE, associated applications, and the tenant network container can be supported through DC orchestration systems.

Unlike a physical or virtual PE which can support multi-tenants, a physical or virtual CE supports a single tenant only. A single tenant CAN use multiple physical or virtual CEs. An end device, such as a server, CAN support one or more vCE(s). While the vCE is defined as a single tenant device, each tenant can have multiple logical departments which are under the tenants administrative control, requiring logical separation, this is the same model as today's physical CE deployments.

Virtual CE and virtual PE are complimentary approaches for extending IP VPN into tenant containers. In the vCE solution, there is no IP VPN within the data center or other type of service network, the vCE can connect to the PE which is a centralized IP VPN PE/Gateway/ASBR, or connect to distributed vPE on a server or on the Top of the Rack switch (ToR). Virtual CE can be used to extend the SP managed CE

solution to create new cloud enabled services and provide the same topological model and features that are consistent with the physical CE systems.

2.2 Architecture

Figure 1 illustrates the topology where vCE is resident in the servers where the applications are hosted.

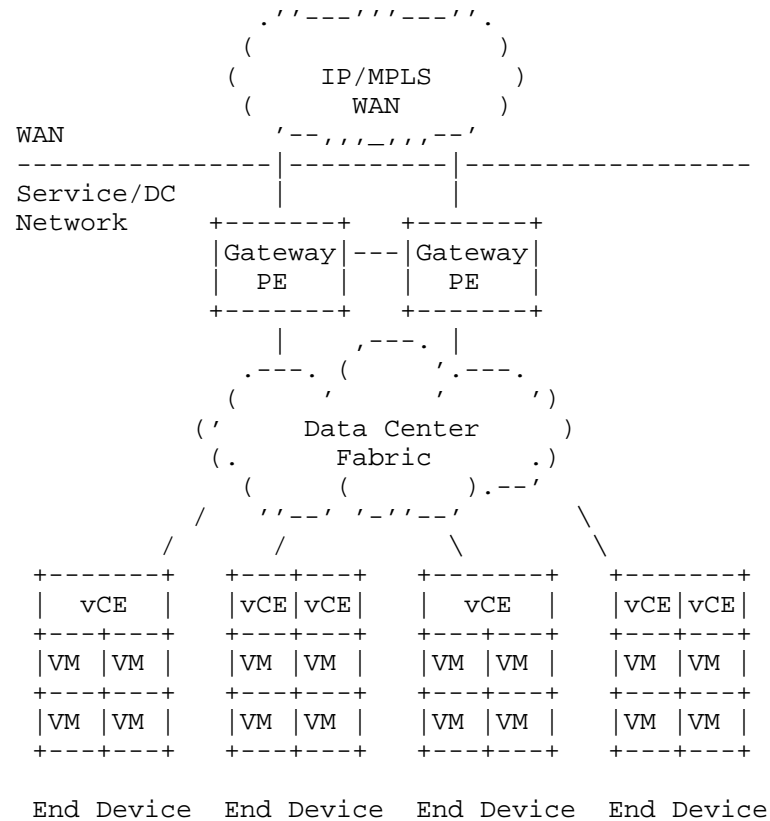


Figure 1. Virtualized Data Center with vCE

Figure 1 shows above vCE solution in a virtualized Data Center with application VMs on the servers. One or more vCEs MAY be used on each server.

The vCEs logically connect to the PEs/Gateway PEs to join the particular IP VPN which the tenant belongs to. Gateway PEs connect to the IP MPLS WAN network for inter-DC and DC to enterprise VPN sites

connection. The server physically connects to the DC Fabric for packet forwarding.

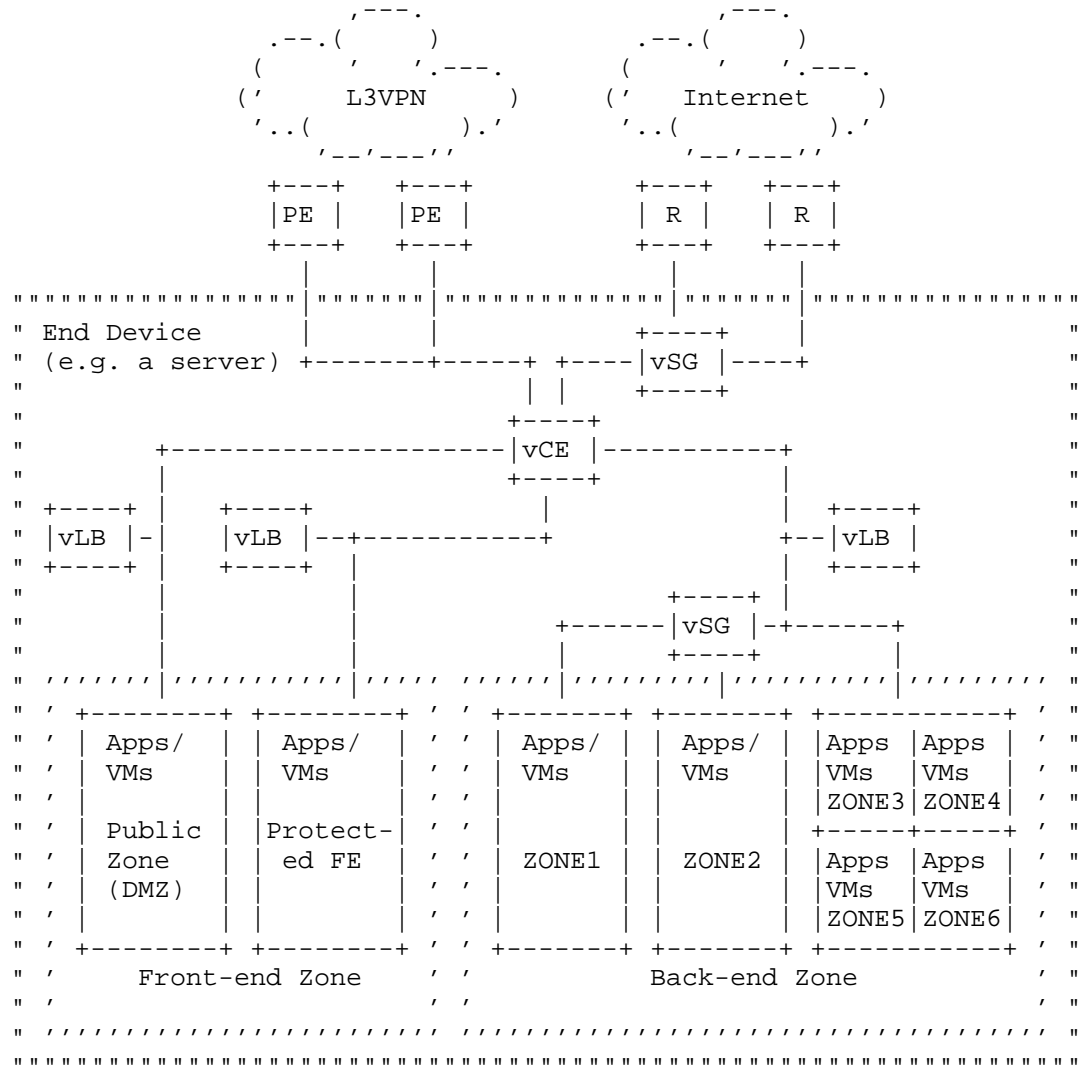


Figure 2. A Virtualized Container with vCE in an End Device

An end device shown in Figure 2 is a physical server supporting multiple virtualized appliances and application, and hosts multiple client VMs. An end device shown in Figure 2 is a physical server supporting multiple In the traditional deployment, the topology often involves multiple physical CEs, physical Security Gateways and Load

Balancers residing in the same Data Center.

The virtualized approach provides the benefit of reduced number of physical devices, simplified management, optimal routing due to the co-location of vCE, services, and client VMs.

While the above diagram represents a simplified view of all of the tenant service and application VMs residing in the same physical server, the above model can also be represented with the VMs spread across many physical servers and the DC network would provide the physical inter-connectivity while the vCE and the VMs connected to the vCE form the logical connections.

3. Control Plane

3.1 vCE Control Plane

The vCE control plane can be distributed or centralized.

1) Distributed control plane

vCE CAN exchange BGP routes with PE or vPE for the particular IP VPN as described in [RFC4364].

The vCE needs to support BGP if this approach is used.

The advantage of distributed protocols is to avoid single point of failure and bottleneck. Service chaining can be easily and efficiently supported in this approach.

BGP as PE-CE protocol is used in about 70% of cases in typical Enterprise IP VPN PE-CE connections. BGP supports rich policy compared to other alternatives.

2) Static routing. It is used in about 30% of cases in Enterprise IP VPN PE-CE connections. It MAY be used if the operator prefers.

2. Using controller approach

Using controller is the Software Defined Network (SDN) approach. A controller can be distributed or centralized. The central controller performs the control plane functions, and sends instructions to the vCE on the end devices to configure the data plane.

This requires standard interface to routing system (I2RS). The Interface to Routing System (I2RS) is work in progress in IETF [I-D.ward-irs-framework], [I-D.rfernando-irs-framework-requirement].

4. Forwarding Plane

4.1 Forwarding between vCE and PE/vPE

No MPLS forwarding is required between PE and CE in typical PE-CE connection scenarios, though MPLS label forwarding is required for implementing Carriers' Carrier (CSC) model.

IPv4 and IPv6 packet forwarding MUST be supported.

Native fabric CAN be used to support isolation between vCEs to PE connections.

Examples of native fabric include:

- VLANs [IEEE 802.1Q], Virtual Local Area Network- IEEE 802.1ad [IEEE 802.1ad]/QinQ, Provider Bridge

Or overlay segmentation with better scalability:

- VXLANs [I-D.mahalingam-dutt-dcops-vxlan], Virtual Extensible LAN- NVGRE [I-D.sridharan-virtualization-nvgre], Network Virtualization using GRE

Note the the above references for overlay network are currently work in progress in IETF.

4.2 Forwarding between vCE and VM

If the vCE and the VM the vCE is connecting are co-located in the same server, the connection is internal to the server, no external protocol involved.

If the vCE and the VM the vCE is connecting are located in different devices, standard external protocols are needed. The forwarding can be native or overlay techniques as listed in the above sub-section.

5. Addressing and QoS

5.1 Addressing

IPv4 and IPv6 addressing MUST be supported.

IP address allocation for vCEs and applications/client:

- 1) IP address MAY be assigned by central management/provisioning with predetermined blocks through planning process.

2) IP address MAY be obtained through DHCP server.

Address space separation: The IP addresses used for clients in the IP VPNs in the Data Center SHOULD be in separate address blocks outside the blocks used for the underlay infrastructure of the Data Center. The purpose is to protect the Data Center infrastructure from being attacked if the attacker gain access of the tenant VPNs.

5.2 QoS

Differentiated Services [RFC2475] Quality of Service (QoS) is standard functionality for physical CEs and MUST be supported on vCE. This is important to ensure seamless end-to-end SLA from IP VPN in the WAN into service network/Data center. The use of MPLS Diffserv tunnel model Pipe Mode (RFC3270) with explicit null LSP must be supported.

6. Management plane

6.1 Network abstraction and management

The use of vCE with single tenant virtual service instances can simplify management requirements as there is no need to discover device capabilities, track tenant dependencies and manage service resources.

vCE North bound interface SHOULD be standards based.

The Interface to Routing System (I2RS) is work in progress in IETF [I-D.ward-irs-framework], [I-D.rfernando-irs-framework-requirement].

vCE element management MUST be supported, it can be in the similar fashion as for physical CE, without the hardware aspects.

6.2 Service VM Management

Service VM Management SHOULD be hypervisor agnostic, e.g. On demand service VMs turning-up SHOULD be supported.

The management tool SHOULD be open standards.

7. Orchestration and IP VPN inter-provisioning

7.1 DC Instance to WAN IP VPN instance "binding" Requirements

- MUST support service activation in the physical and virtual environment.

For example, assign VLAN to correct VRF.

- MUST support per VLAN Authentication, Authorization, and Accounting (AAA).

The PE function is an OA&M boundary.

- MUST be able to apply other policies to VLAN.

For example, per VLAN QOS, ACLs.

- MUST ensure that WAN IP VPN state and Data cCentre state are dynamically synchronized.

Ensure that there is no possibility of customer being connected to the wrong VRF. For example, remove all tenant state when service instance terminated.

- MUST integrate with existing WAN IP VPN provisioning processes.
- MUST scale to at least 10,000 tenant service instances.
- MUST cope with rapid (sub minute) tenant mobility.
- MAY support Automated cross provisioning accounting correlation between WAN IP VPN and cloud/DC for the same tenant.
- MAY support Automated cross provisioning state correlation between WAN IP VPN and cloud/DC/extended Data Center for the same tenant.

7.2. Provisioning/Orchestration

There are two primary approaches for IP VPN provisioning - push and pull, both CAN be used for provisioning/orchestration.

7.2.1 vCE Push model

Push model: It is a top down approach - push IP VPN provisioning from network management system or other central control provisioning systems to the IP VPN network elements.

This approach supports service activation and it is commonly used in the existing IP VPN enterprise deployment. When existing the IP VPN solution into the cloud/data center or separate Data Center, it MUST support off-line accounting correlation between the WAN IP VPN and the cloud/DC IP VPN for the tenant, the systems SHOULD be able to bind interface accounting to particular tenant. It MAY requires

offline state correlation as well, for example, bind interface state to tenant.

7.2.1.1 Inter-domain provisioning vCE Push Model

Provisioning process:

- 1) Cloud/DC orchestration configures vCE.
- 2) Orchestration initiates WAN IP VPN provisioning; passes connection IDs (e.g., of VLAN/VXLAN) and tenant context to WAN IP VPN provisioning systems.
- 3) WAN IP VPN provisioning system provisions PE VRF and other policies per normal enterprise IP VPN provisioning processes.

This model requires the following:

- The DC Orchestration system or the WAN IP VPN provisioning system know the topology inter-connecting the DC and WAN VPN. For example, which interface on the WAN core device connects to which interface on the DC PE.
- Offline state correlation.
- Offline accounting correlation.
- Per SP integration.

Dynamic BGP session between PE/vPE and vCE MAY be used to automate the PE provisioning in the PE-vCE model, that will remove the needs for PE configuration. Other protocols can be used for this purpose as well, for example, use Enhanced Interior Gateway Routing Protocol (EIGRP) for dynamic neighbour relationship establishment.

The dynamic routing Prevents the need to configure the PEs in PE-vCE model.

Caution: This is only under the assumption that the DC provisioning system is trusted and could support dynamic establishment of PE-vCE BGP neighbor relationships, for example, the WAN network and the cloud/DC belongs to the same Service Provider.

7.2.1.2 Cross-domain provisioning vCE Push Model

Provisioning Process:

- 1) Cross-domain orchestration system initiates DC orch.

- 2) DC orchestration system configures vCE
- 3) DC orchestration system passes back VLAN/VXLAN and tenant context to Cross-domain orchestration system
- 4) Cross-domain orchestration system initiates WAN IP VPN provisioning
- 5) WAN IP VPN provisioning system provisions PE VRF and other policies as per normal enterprise IP VPN provisioning processes.

This model requires the following:

- Cross-domain orchestration system knows the topology connecting the DC and WAN IP VPN, for example, which interface on core device connects to which interface on DC PE.- Offline state correlation.
- Offline accounting correlation.
- Per SP integration.

7.1.1 vCE Pull model

Pull model: It is a bottom-up approach - pull from network elements to network management/AAA based upon data plane or control plane activity. It supports service activation, this approach is often used in broadband deployment. Dynamic accounting correlation and dynamic state correlation are supported. For example, session based accounting is implicitly includes tenant context state correlation, as well as session based state which implicitly includes tenant context.

Inter-domain Provisioning:

Process:

- 1) Cloud/DC orchestration system configures vCE
- 2) Cloud/DC Orchestration system primes WAN IP VPN provisioning/AAA for new service, passes connection IDs (e.g., VLAN/VXLAN) and tenant context WAN IP VPN provisioning systems.
- 3) Cloud/DC PE detects new VLAN, send Radius Access-Request.
- 4) Radius Access-Accept with VRF and other policies.

This model requires VLAN/VLAN information and tenant context to passed on a per transaction basis. In practice, it may simplify to

use DC orchestration updating LDAP directory

Auto accounting correlation and auto state correlation is supported.

8. vCE and vPE interaction

A vPE ([I-D.fang-l3vpn-virtual-pe] [I-D.ietf-l3vpn-end-system]) is treating the VMs in the server as a virtual CE. In this section, the relationship between the vPE and such vCE is discussed. vPE can support one of the following two models:

Model 1: a limited control-plane functionality that advertises local VPN routes to a controller and receive VPN routes from the controller.

Model 2: a control plane component physically separated from the forwarding component that fully performs the control plane routing functionality and communicate FIB entries to the vPE forwarding entity implemented on servers.

A vCE provides subnet routing, firewalling or SLB services to host VMs. The underlying connectivity between the vCE and these VMs can be at layer 2 or layer 3. In addition, the vCE can be connected to other vCEs over Layer 2 or using an IP VPN infrastructure. In this section, the focus is on IP VPN connectivity and more importantly on the interaction between a vCE, a traditional PE (simply referred to as PE), and between a vCE and a vPE.

8.1 Traditional vCE-PE connectivity

This connectivity is described in BGP/MPLS IPVPN [RFC4364]. The only distinction being that the VE is a virtual CE. The vCE attaches to the layer 3 PE using a layer2 logical connection, e.g., Ethernet VLAN, or a tunnel (e.g., IP/GRE, VXLAN) that are presented as IP interfaces to a corresponding VRF at the PE. Routing between the vCE and PE can be static or based on a dynamic routing protocol (e.g., OSPF, BGP). A routing protocol, in addition to enabling the exchange of routing information between the PE and vCE, provides liveness check between the vCE and the PE. In the absence of a dynamic routing protocol, the vCE must support a mechanism that provides for liveness check, or an out-of-band mechanism must be implemented to monitor the liveness of a vCE and a connected PE, and effect routing changes upon a failure. Options for in-band liveness check include IP BFD [RFC5880], Ethernet Continuity Check (CC) [IEEE 802.1ag], and IP ping [RFC4560]. IP BFD must be supported while the other mechanisms are optional.

8.2 vCE-vPE connectivity

In this model, the vCE and vPE forwarding plane can be: (1) co-located on the same end device, e.g., a server, or (2) located on different servers. In addition, the control plane interaction differs between vPE model 1 and model 2.

8.2.1 Co-located vCE-vPE connectivity with vPE Model 1

In vPE Model 1, there is a control plane component of the vPE implemented on the end-server (e.g., [I-D.ietf-l3vpn-end-system], [I-D.fang-l3vpn-virtual-pe]). In addition, there is a control plane component implemented on a separate control plane entity (out-of-band) that enables the exchange of routing information among vPEs. In [I-D.ietf-l3vpn-end-system], the out-of-band control plane component is referred to as router server; in [I-D.fang-l3vpn-virtual-pe], it is referred to as vPE-C. There are two cases that must be considered:

Case 1-A: vCE to vPE local route exchange on a server / vPE-C

Case 1-B: vCE to route server / vPE-C route exchange.

In these two cases, the vPE control plane or route server must send the CE a default route with next hop being the co-located vPE forwarding plane entity.

In case 1-A, the vCE must send local routes to the vPE control plane with itself being the next hop. The vPE control plane entity in turn updates the out-of-band control entity (e.g., route server) with routes reachable via the local CE, as VPN routes, with itself being the next hop for these routes. The vPE also receives from the route server VPN routes reachable via other vPEs [end-system]. It should be noted in this case, that the vCE must be able support one or more routing contexts, each with separate attachment circuit to the vPE. Each such routing context must be associated with a VPN and one or more VPNs must be supported.

In case 1-B, the vCE must have a control channel with a route server. There must be a control channel per vCE routing context or alternatively must allow the unambiguous multiplexing of routes that belong to different routing context on the same channel. The vCE sends routes reachable via the vCE to the route server with itself being the next hop. The route server must learn from the co-located vPE control plane component reachability to the local vCE IP address used as next hop. This IP address must be exchanged between the vCE and vPE in-band over a corresponding attachment circuit that identifies the routing context. Alternatively, the route server/vPE-C must be programmed with the association of the vCE control channel, a VPN and an end-device IP address. As a result, the route server/vPE-C must populate the vPE distributed control plane with the

corresponding routes as non-VPN routes and the vPE must respond with VPN routes that correspond to each of these routes. Alternatively, routes reachable via a vCE must be defined via in portal per routing context and therefore VPN, and then correlated upon instantiation of the vCE on an end-system with the end-system IP address and the appropriate VRF on that end-system. In addition, the vCE must be configured with default routes per routing context with the next hop being the vPE.

8.2.2 Co-located vCE-vPE connectivity with vPE Model 2

In this model, there is no control plane routing component implemented on the end-system. That, is the end-system does not generate VPN routes and only receives VPN FIB entries from the out-of-band control plane component for routes reachable locally and for remote routes. The vCE-control plane interaction is similar to that of the interaction in Model 1 case 1-B described in the previous section whereby route population is management-driven.

8. Security Considerations

vCE creation on server - is server owned by the the operator? is this managed CE model? how to authenticate?

vCE in DC connecting VPN in WAN IP - are the DC and WAN IP VPN belong to the same SP or different? How much info are permitted to pass through auto-provisioning? How to authenticate connections, especially in pull models?

How vCE protects itself from attach from client VMs?

Additional security procedures in all virtualized cloud/DC environment, FW placement. All virtualized appliances need to be protected against attack.

Three tier (Web, App, DB) interaction access control.

Details to be added.

9. IANA Considerations

None.

10. References

10.1 Normative References

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

Requirement Levels", BCP 14, RFC 2119, March 1997.

- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [RFC4560] Quittek, J., Ed., and K. White, Ed., "Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations", RFC 4560, June 2006.
- [RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.
- [I-D.ietf-l3vpn-end-system] Marques, P., Fang, L., Pan, P., Shukla, A., Napierala, M., "BGP-signaled end-system IP/VPNs", draft-ietf-l3vpn-end-system, work in progress.
- [IEEE 802.1ad] IEEE, "Provider Bridges", 2005.
- [IEEE 802.1q] IEEE, "802.1Q - Virtual LANs", 2006.
- [IEEE 802.1ag] IEEE "802.1ag - Connectivity Fault Management", 2007.

10.2 Informative References

- [RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Service", RFC 2475, December 1998.
- [I-D.fang-l3vpn-virtual-pe] Fang, L., Ward, D., Fernando, R., Napierala, M., Bitar, N., Rao, D., Rijsman, B., So, N., "BGP IP VPN Virtual PE", draft-fang-l3vpn-virtual-pe, work in progress.
- [I-D.ward-irs-framework] Atlas, A., Nadeau, T., Ward, D., "Interface to the Routing System Framework", draft-ward-irs-framework, work in progress.

- [I-D.rfernando-irs-framework-requirement] Fernando, R., Medved, J., Ward, D., Atlas, A., Rijsman, B., "IRS Framework Requirements", draft-rfernando-irs-framework-requirement-00, work in progress.
- [I-D.mahalingam-dutt-dcops-vxlan]: Mahalingam, M, Dutt, D., et al., "A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks" draft-mahalingam-dutt-dcops-vxlan, work in progress.
- [I-D.sridharan-virtualization-nvgre]: SridharanNetwork, M., et al., "Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre,, work in progress.

11. Acknowledgement

The authors would like to thank Vaughn Suazo for his review and comments.

Authors' Addresses

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

John Evans
Cisco
16-18 Finsbury Circus
London, EC2M 7EB
UK
Email: joevans@cisco.com

David Ward
Cisco
170 W Tasman Dr
San Jose, CA 95134
US
Email: wardd@cisco.com

Rex Fernando
Cisco
170 W Tasman Dr
San Jose, CA

US
Email: rex@cisco.com

Ning So
Vinci Systems
Email: ning.so@vinci-systems.com

Nabil Bitar
Verizon
40 Sylvan Road
Waltham, MA 02145
Email: nabil.bitar@verizon.com

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

INTERNET-DRAFT
Intended Status: Standards track
Expires: January 4, 2015

Ning So
Vinci Systems
Jim Guichard
Cisco
Wen Wang
CenturyLink
Manuel Paul
Deutsche Telekom
Wim Henderichx
Alcatel-Lucent

Luyuan Fang, Ed.
Microsoft
David Ward
Rex Fernando
Cisco
Maria Napierala
AT&T
Nabil Bitar
Verizon
Dhananjaya Rao
Cisco
Bruno Rijsman
Juniper

July 4, 2014

BGP/MPLS VPN Virtual PE
draft-fang-l3vpn-virtual-pe-05

Abstract

This document describes the architecture solutions for BGP/MPLS L3 and L2 Virtual Private Networks (VPNs) with virtual Provider Edge (vPE) routers. It provides a functional description of the vPE control, forwarding, and management. The proposed vPE solutions support both the Software Defined Networks (SDN) approach which allows physical decoupling of the control and the forwarding, and the traditional distributed routing approach. A vPE can reside in any network or compute devices, such as a server as co-resident with the application virtual machines (VMs), or a Top-of-Rack (ToR) switch in a Data Center (DC) network.

Status of this Memo

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

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

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

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

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

Copyright and License Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

1	Introduction	4
1.1	Terminology	4
1.2	Requirements	5
2.	Virtual PE Architecture	6
2.1	Virtual PE definitions	6
2.2	vPE Architecture and Design options	8
2.2.1	vPE-F host location	8
2.2.2	vPE control plane topology	8
2.2.3	Data Center orchestration models	8
2.3	vPE Architecture reference models	8
2.3.1	vPE-F in an end-device and vPE-C in the controller	8
2.3.2	vPE-F and vPE-C on the same end-device	10
2.3.3	vPE-F and vPE-C are on the ToR	11
2.3.4	vPE-F on the ToR and vPE-C on the controller	12
2.3.5	The server view of a vPE	12
3.	Control Plane	13
3.1	vPE Control Plane (vPE-C)	13
3.1.1	The SDN approach	13
3.1.2	Distributed control plane	14
3.3	Use of router reflector	14
3.4	Use of Constrained Route Distribution [RFC4684]	14
4.	Forwarding Plane	14
4.1	Virtual Interface	14
4.2	Virtual Provider Edge Forwarder (vPE-F)	15

4.3 Encapsulation	15
4.4 Optimal forwarding	15
4.5 Routing and Bridging Services	16
5. Addressing	17
5.1 IPv4 and IPv6 support	17
5.2 Address space separation	17
6.0 Inter-connection considerations	17
7. Management, Control, and Orchestration	18
7.1 Assumptions	18
7.2 Management/Orchestration system interfaces	19
7.3 Service VM Management	19
7.4 Orchestration and MPLS VPN inter-provisioning	19
7.4.1 vPE Push model	20
7.4.2 vPE Pull model	21
8. Security Considerations	21
9. IANA Considerations	22
10. Acknowledgments	22
11. References	22
11.1 Normative References	22
11.2 Informative References	23
Authors' Addresses	24

1 Introduction

Network virtualization enables multiple isolated individual networks over a shared common network infrastructure. BGP/MPLS IP Virtual Private Networks (IP VPNs) [RFC4364] have been widely deployed to provide network based Layer 3 VPNs solutions. [RFC4364] provides routing isolation among different customer VPNs and allow address overlap among these VPNs through the implementation of per VPN Virtual Routing and Forwarding instances (VRFs) at a Service Provider Edge (PE) routers, while forwarding customer traffic over a common IP/MPLS network. For L2 VPN, a similar technology is being defined in [I-D.ietf-l2vpn-evpn] on the basis of BGP/MPLS, to provide switching isolation and allow MAC address overlap.

With the advent of compute capabilities and the proliferation of virtualization in Data Center servers, multi-tenant Data Centers are becoming the norm. As applications and appliances are increasingly being virtualized, support for virtual edge devices, such as virtual L3/L2 VPN PE routers, becomes feasible and desirable for Service Providers who want to extend their existing MPLS VPN deployments into Data Centers to provide end-to-end Virtual Private Cloud (VPC) services. Virtual PE work is also one of early effort for Network Functions Virtualization (NFV). In general, scalability, agility, and cost efficiency are primary motivations for vPE solutions.

The virtual Provider Edge (vPE) solution described in this document allows for the extension of the PE functionality of L3/L2 VPN to an end device, such as a server where the applications reside, or to a first hop routing/switching device, such as a Top of the Rack (ToR) switch in a DC.

The vPE solutions support both the Software Defined Networks (SDN) approach, which allows physical decoupling of the control and the forwarding, and the traditional distributed routing approach.

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].

Term	Definition
ASBR	Autonomous System Border Router
BGP	Border Gateway Protocol
CE	Customer Edge
Forwarder	IP VPN forwarding function

GRE	Generic Routing Encapsulation
Hypervisor	Virtual Machine Manager
I2RS	Interface to Routing Systems
LDP	Label Distribution Protocol
MP-BGP	Multi-Protocol Border Gateway Protocol
MPLS	Multi-Protocol Label Switching
PCEF	Policy Charging and Enforcement Function
QoS	Quality of Service
RR	Route Reflector
RT	Route Target
RTC	RT Constraint
SDN	Software Defined Networks
ToR	Top-of-Rack switch
VI	Virtual Interface
vCE	virtual Customer Edge Router
VM	Virtual Machine
vPC	virtual Private Cloud
vPE	virtual Provider Edge Router
vPE-C	virtual Provider Edge Control plane
vPE-F	virtual Provider Edge Forwarder
VPN	Virtual Private Network
vRR	virtual Route Reflector
WAN	Wide Area Network

End device: where Guest OS, Host OS/Hypervisor, applications, VMs, and virtual router may reside.

1.2 Requirements

The following are key requirements for vPE solutions.

- 1) MUST support end device multi-tenancy, per tenant routing isolation and traffic separation.
- 2) MUST support large scale MPLS VPNs in the Data Center, upto tens of thousands of end devices and millions of VMs in the single Data Center.
- 3) MUST support end-to-end MPLS VPN connectivity, e.g. MPLS VPN can start from a DC end device, connect to a corresponding MPLS VPN in the WAN, and terminate in another Data Center end device.
- 4) MUST allow physical decoupling of MPLS VPN PE control and forwarding for network virtualization and abstraction.
- 5) MUST support the control plane with both SDN controller approach, and the traditional distributed control plane approach with MP-BGP protocol.

- 6) MUST support VM mobility.
- 7) MUST support orchestration/auto-provisioning deployment model.
- 8) SHOULD be capable to support service chaining as part of the solution [I-D.rfernando-l3vpn-service-chaining], [I-D.bitars-i2rs-service-chaining].

The architecture and protocols defined in BGP/MPLS IP VPN [RFC4364] and BGP/MPLS EVPN [I-D.ietf-l2vpn-evpn] provide the foundation for vPE extension. Certain protocol extensions may be needed to support the virtual PE solutions.

2. Virtual PE Architecture

2.1 Virtual PE definitions

As defined in [RFC4364] and [I-D.ietf-l2vpn-evpn], an MPLS VPN is created by applying policies to form a subset of sites among all sites connected to the backbone networks. It is a collection of "sites". A site can be considered as a set of IP/ETH systems maintaining IP/ETH inter-connectivity without direct connecting through the backbone. The typical use of L3/L2 VPN has been to inter-connect different sites of an Enterprise networks through a Service Provider's BGP MPLS VPNs in the WAN.

A virtual PE (vPE) is a BGP/MPLS L3/L2 VPN PE software instance which may reside in any network or computing devices. The control and forwarding components of the vPE can be decoupled, they may reside in the same physical device, or in different physical devices.

A virtualized Provider Edge Forwarder (vPE-F) is the forwarding element of a vPE. vPE-F can reside in an end device, such as a server in a Data Center where multiple application Virtual Machines (VMs) are supported, or a Top-of-Rack switch (ToR) which is the first hop switch from the Data Center edge. When a vPE-F is residing in a server, its connection to a co-resident VM can be viewed as similar to the PE-CE relationship in the regular BGP L3/L2 VPNs, but without routing protocols or static routing between the virtual PE and end-host because the connection is internal to the device.

The vPE Control plane (vPE-C) is the control element of a vPE. When using the approach where control plane is decoupled from the physical topology, the vPE-F may be in a server and co-resident with application VMs, while one vPE-C can be in a separate device, such as an SDN Controller where control plane elements and orchestration functions are located. Alternatively, the vPE-C can reside in the same physical device as the vPE-F. In this case, it is similar to the

traditional implementation of VPN PEs where, distributed MP-BGP is used for L3/L2 VPN information exchange, though the vPE is not a dedicated physical entity as it is in a physical PE implementation.

2.2 vPE Architecture and Design options

2.2.1 vPE-F host location

Option 1a. vPE-F is on an end device as co-resident with application VMs. For example, the vPE-F is on a server in a Data Center.

Option 1b. vPE-F forwarder is on a ToR or other first hop devices in a DC, not as co-resident with the application VMs.

Option 1c. vPE-F is on any network or compute devices in any types of networks.

2.2.2 vPE control plane topology

Option 2a. vPE control plane is physically decoupled from the vPE-F. The control plane may be located in a controller in a separate device (a stand alone device or can be in the gateway as well) from the vPE forwarding plane.

Option 2b. vPE control plane is supported through dynamic routing protocols and located in the same physical device as the vPE-F.

2.2.3 Data Center orchestration models

Option 3a. Push model: It is a top down approach, push IP VPN provisioning state from a network management system or other centrally controlled provisioning system to the IP VPN network elements.

Option 3b. Pull model: It is a bottom-up approach, pull state information from network elements to network management/AAA based upon data plane or control plane activity.

2.3 vPE Architecture reference models

2.3.1 vPE-F in an end-device and vPE-C in the controller

Figure 1 illustrates the reference model for a vPE solution with the vPE-F in the end device co-resident with applications VMs, while the vPE-C is physically decoupled and residing on a controller.

The Data Center is connected to the IP/MPLS core via the Gateways/ASBRs. The MPLS VPN, e.g. VPN RED, has a single termination point within the DC at one of the VPE-F, and is inter-connected in the WAN to other member sites which belong to the same client, and the remote ends of VPN RED can be a PE which has VPN RED attached to it, or another vPE in a different Data Center.

Note that the DC fabrics/intermediate underlay devices in the DC do not participate IP VPNs, their function is the same as provider backbone routers in the IP/MPLS back bone and they do not maintain the VPN states, nor they are VPN aware.

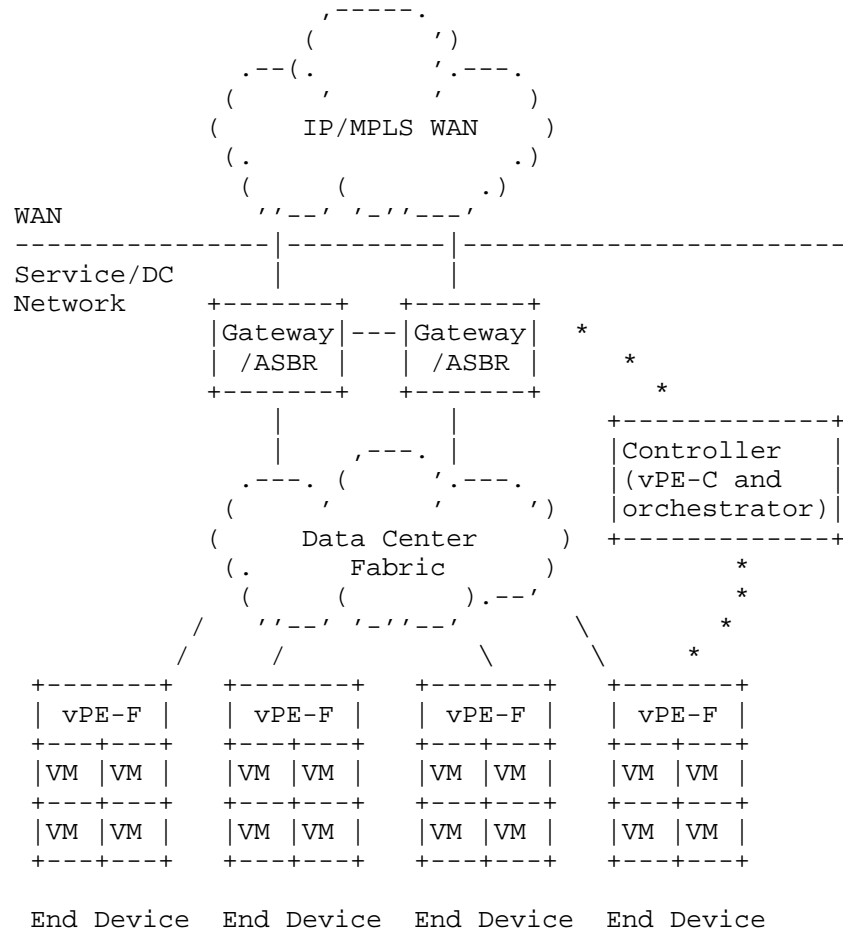


Figure 1. Virtualized Data Center with vPE at the end device and vPE-C and vPE-F physically decoupled

Note:

- *** represents Controller logical connections to the all Gateway/ASBRs and to all vPE-F.
- ToR is assumed included in the Data Center cloud.

2.3.2 vPE-F and vPE-C on the same end-device

In this option, vPE-F and vPE-C functionality are both resident in the end-device. The vPE functions the same as it is in a physical PE. MP-BGP is used for the VPN control plane. Virtual or physical Route Reflectors (RR) (not shown in the diagram) can be used to assist scaling.

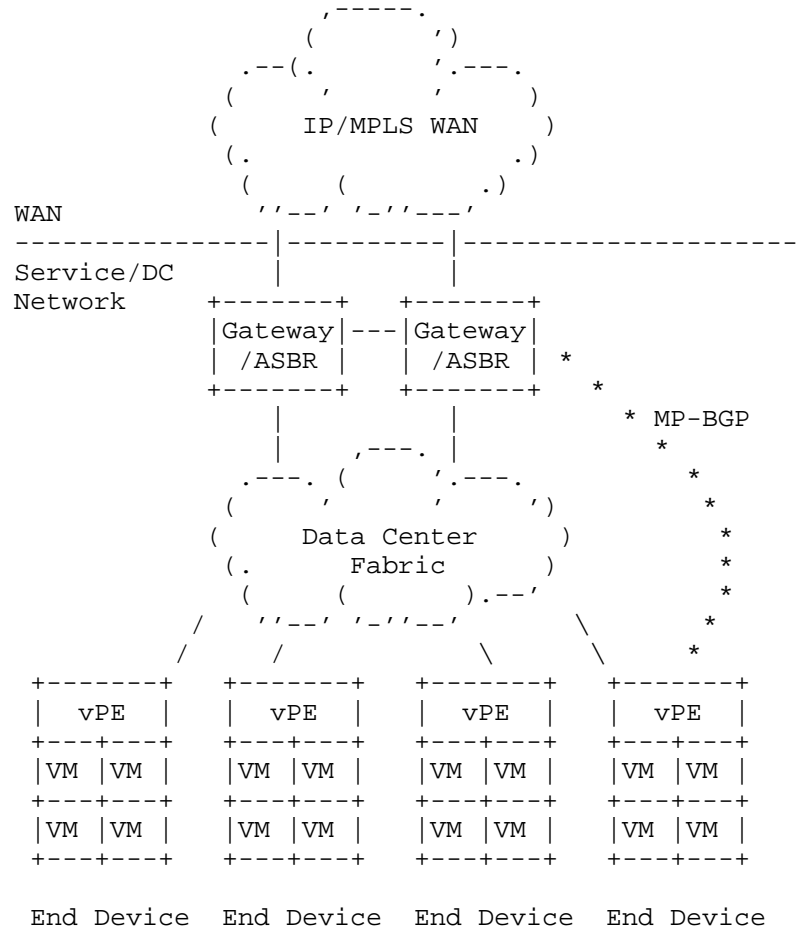


Figure 2. Virtualized Data Center with vPE at the end device, VPN control signal uses MP-BGP

Note:

a) *** represents the logical connections using MP-BGP among the Gateway/ASBRs and to the vPEs on the end devices.

b) ToR is assumed included in the Data Center cloud.

2.3.3 vPE-F and vPE-C are on the ToR

In this option, vPE functionality is the same as a physical PE. MP-BGP is used for the VPN control plane. Virtual or physical Route Reflector (RR) (not shown in the diagram) can be used to assist scaling.

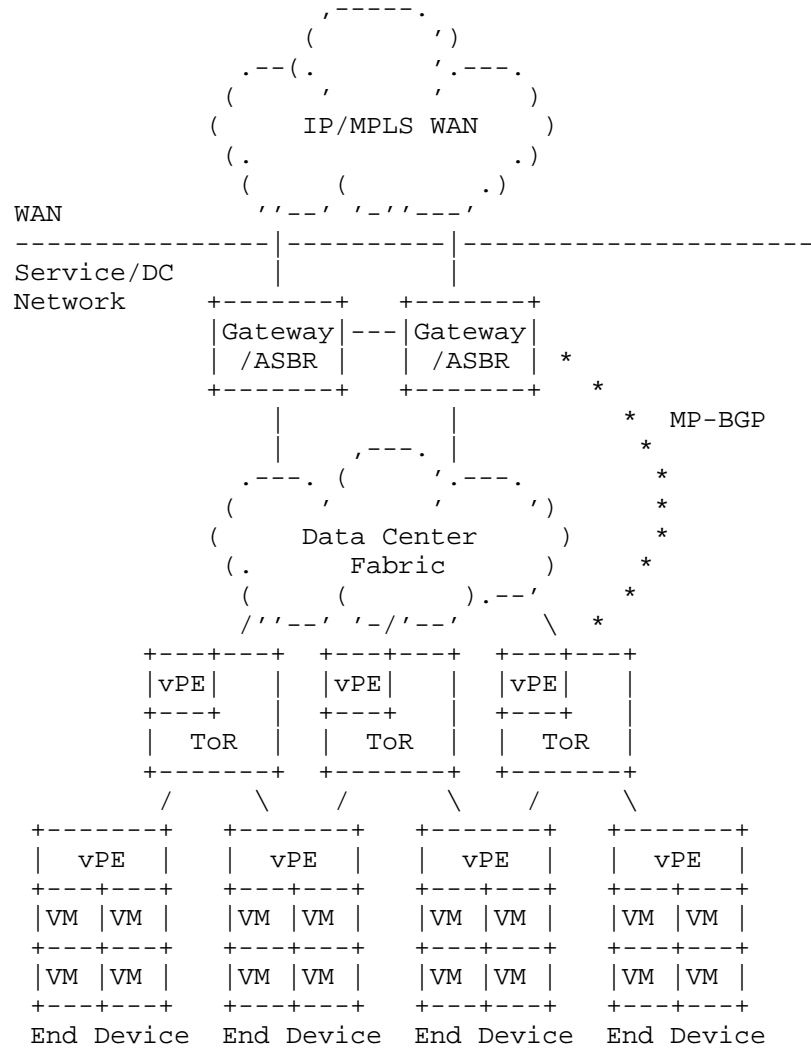


Figure 3. Virtualized Data Center with vPE at the ToP, VPN control signal uses MP-BGP

Note: *** represents the logical connections using MP-BGP among the Gateway/ASBRs and to the vPEs on the ToRs.

2.3.4 vPE-F on the ToR and vPE-C on the controller

In this option, the L3/L2 VPN termination is at the ToR, but the control plane decoupled from the data plane and resided in a controller, which can be on a stand alone device, or can be placed at the Gateway/ASBR.

2.3.5 The server view of a vPE

An end device shown in Figure 4 is a virtualized server that hosts multiple VMs. The virtual PE is co-resident in the server with application VMs. The vPE supports multiple VRFs, VRF Red, VRF Grn, VRF Yel, VRF Blu, etc. Each application VM is associated to a particular VRF as a member of the particular VPN. For example, VM1 is associated to VRF Red, VM2 and VM47 are associated to VRF Grn, etc. Routing/switching isolation applies between VPNs for multi-tenancy support. For example, VM1 and VM2 cannot communicate directly in a simple intranet VPN topology as shown in the configuration.

The vPE connectivity relationship between vPE and the application VM is similar to the PE-to-CE relationship in regular BGP VPNs. However, as the vPE and end-host functions are co-resident in the same server, the connection between them is an internal implementation of the server.

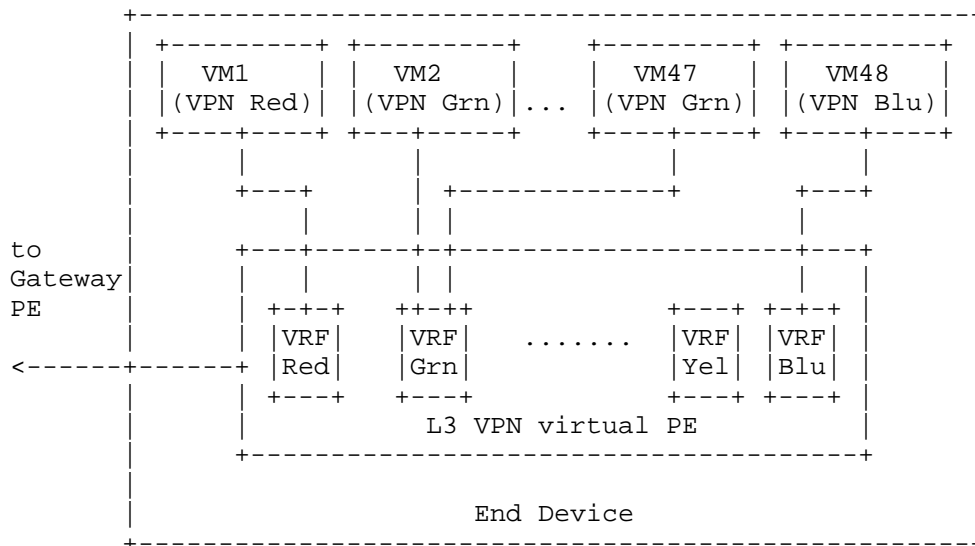


Figure 4. Server View of vPE to VM relationship

An application VM may send packets to a vPE forwarder that need to be bridged, either locally to another VM, or to a remote destination. In this case, the vPE contains a virtual bridge instance to which the application VMs (CEs) are attached.

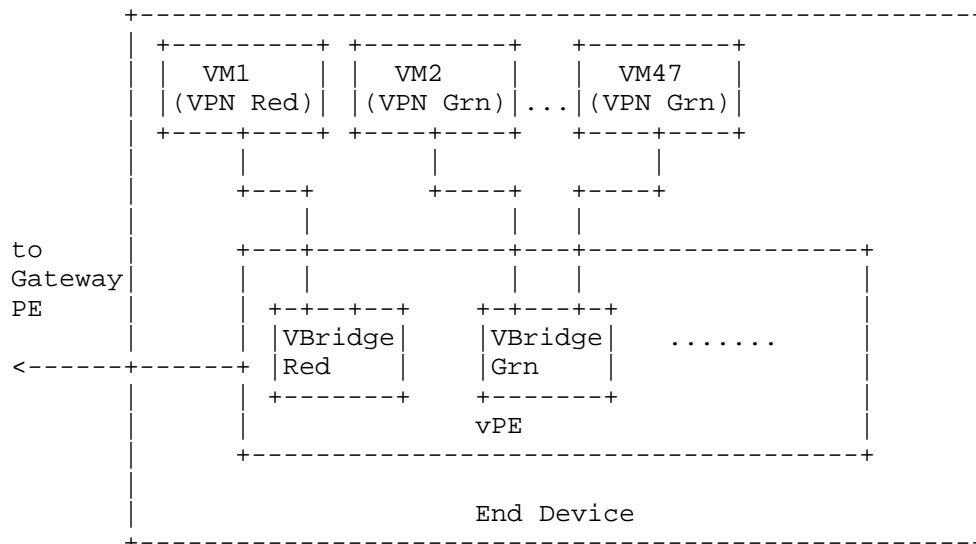


Figure 4. Bridging Service at vPE

3. Control Plane

3.1 vPE Control Plane (vPE-C)

3.1.1 The SDN approach

This approach is appropriate when the vPE control and data planes are physically decoupled. The control plane directing the data flow may reside elsewhere, e.g. in a SDN controller. This approach requires a standard interface to the routing system. The Interface to Routing System (I2RS) is work in progress in IETF as described in [I-D.ietf-i2rs-architecture], [I-D.ietf-i2rs-problem-statement].

Although MP-BGP is often the de facto preferred choice between vPE and gateway-PE/ASBR, the use of extensible signaling messaging protocols MAY often be more practical in a Data Center environment. One such proposal that uses this approach is detailed in [I-D.ietf-l3vpn-end-system].

3.1.2 Distributed control plane

In the distributed control plane approach, the vPE participates in the overlay L3/L2 VPN control protocol: MP-BGP [RFC4364].

When the vPE function is on a ToR, it participates the underlay routing through IGP protocols (ISIS or OSPF) or BGP.

When the vPE function is on a server, it functions as a host attached to a server.

3.3 Use of router reflector

Modern Data Centers can be very large in scale. For example, the number of VPNs routes in a very large DC can surpass the scale of those in a Service Provider backbone VPN networks. There may be tens of thousands of end devices in a single DC.

Use of Router Reflector (RR) is necessary in large-scale IP VPN networks to avoid a full iBGP mesh among all vPEs and PEs. The VPN routes can be partitioned to a set of RRs, the partitioning techniques are detailed in [RFC4364] and [I-D.ietf-l2vpn-evpn].

When a RR software instance is residing in a physical device, e.g., a server, which is partitioned to support multi-functions and application VMs, the RR becomes a virtualized RR (vRR). Since RR performs control functions only, a dedicated or virtualized server with large scale of computing power and memory can be a good candidate as host of vRRs. The vRR can also reside in a Gateway PE/ASBR, or in an end device.

3.4 Use of Constrained Route Distribution [RFC4684]

The Constrained Route Distribution [RFC4684] is a powerful tool for selective VPN route distribution. With RTC, only the BGP receivers (e.g, PE/vPE/RR/vRR/ASBRs, etc.) with the particular IP VPNs attached will receive the route update for the corresponding VPNs. It is critical to use constrained route distribution to support large-scale IP VPN developments.

4. Forwarding Plane

4.1 Virtual Interface

A Virtual Interface (VI) is an interface within an end device that is used for connection of the vPE to the application VMs in the same end device. Such application VMs are treated as CEs in the regular VPN's view.

4.2 Virtual Provider Edge Forwarder (vPE-F)

The Virtual Provider Edge Forwarder (vPE-F) is the forwarding component of a vPE where the tenant identifiers (for example, MPLS VPN labels) are pushed/popped.

The vPE-F location options include:

- 1) Within the end device where the virtual interface and application VMs are located.
- 2) In an external device such as a Top of the Rack switch (ToR) in a DC into which the end device connects.

Multiple factors should be considered for the location of the vPE-F, including device capabilities, overall solution economics, QoS/firewall/NAT placement, optimal forwarding, latency and performance, operational impact, etc. There are design tradeoffs, it is worth the effort to study the traffic pattern and forwarding looking trend in your own unique Data Center as part of the exercise.

4.3 Encapsulation

BGP/MPLS VPNs can be tunneled through the network as overlays using MPLS-based or IP-based encapsulation.

In the case of MPLS-based encapsulation, most existing core deployments use distributed protocols such as Label Distribution Protocol (LDP), [RFC3032][RFC5036], or RSVP-TE [RFC3209].

Due to its maturity, scalability, and header efficiency, MPLS Label Stacking is gaining traction by service providers, and large-scale cloud providers in particular, as the unified forwarding mechanism of choice.

With the emergence of the SDN paradigm, label distribution may be achieved through SDN controllers, or via a combination of centralized control and distributed protocols.

In the case of IP-based encapsulation, MPLS VPN packets are encapsulated in IP or Generic Routing Encapsulation (GRE), [RFC4023], [RFC4797]. IP-based encapsulation has not been extensively deployed for BGP/MPLS VPN in the core; however it is considered as one of the tunneling options for carrying MPLS VPN overlays in the data center. Note that when IP encapsulation is used, the associated security properties must be analyzed carefully.

4.4 Optimal forwarding

Many large cloud service providers have reported the DC traffic is now dominated by East-West across subnet traffic (between the end device hosting different applications in different subnets) rather than North-South traffic (going in/out of the Data Center and to/from the WAN) or switched traffic within subnets. This is the primary reason that newer DC design has moved away from traditional Layer-2 design to Layer-3, especially for the overlay networks.

When forwarding the traffic within the same VPN, the vPE SHOULD be capable to provide direct communication among the VMs/application senders/receivers without the need of going through Gateway devices. If the senders and the receivers are on the same end device, the traffic SHOULD NOT need to leave the device. If they are on different end devices, optimal routing SHOULD be applied.

Extranet MPLS VPN techniques can be used for multiple VPNs access without the need of Gateway facilitation. This is done through the use of VPN policy control mechanisms.

In addition, ECMP is a built in IP mechanism for load sharing. Optimal use of available bandwidth can be achieved by virtue of using ECMP in the underlay, as long as the encapsulation includes certain entropy in the header, VXLAN is such an example.

4.5 Routing and Bridging Services

A VPN forwarder (vPE-F) may support both IP forwarding as well as Layer 2 bridging for traffic from attached end hosts. This traffic may be between end hosts attached to the same VPN forwarder or to different VPN forwarders.

In both cases, forwarding at a VPN forwarder takes place based on the IP or MAC entries provisioned by the vPE controller.

When the vPE is providing Layer 3 service to the attached CEs, the VPN forwarder has a VPN VRF instance with IP routes installed for both locally attached end-hosts and ones reachable via other VPN forwarders. The vPE may perform IP routing for all IP packets in this mode.

When the vPE provides Layer 2 service to the attached end-hosts, the VPN forwarder has an E-VPN instance with appropriate MAC entries.

The vPE may support an Integrated Routing and Bridging service, in which case the relevant VPN forwarders will have both MAC and IP table entries installed, and will appropriately route or switch incoming packets.

The vPE controller performs the necessary provisioning functions to support various services, as defined by an user.

5. Addressing

5.1 IPv4 and IPv6 support

IPv4 and IPv6 MUST be supported in the vPE solution.

This may present a challenge for older devices, but this normally is not an issue for the newer generation of forwarding devices and servers. Note that a server is replaced much more frequently than a network router/switch, and newer equipment SHOULD be capable of IPv6 support.

5.2 Address space separation

The addresses used for the IP VPN overlay in a DC, SHOULD be taken from separate address blocks outside the ones used for the underlay infrastructure of the DC. This practice is to protect the DC infrastructure from being attacked if the attacker gains access to the tenant VPNs.

Similarity, the addresses used for the DC SHOULD be separated from the WAN backbone addresses space.

6.0 Inter-connection considerations

The inter-connection considerations in this section are focused on intra-DC inter-connections.

There are deployment scenarios where BGP/MPLS IP VPN may not be supported in every segment of the networks to provide end-to-end IP VPN connectivity. A vPE may be reachable only via an intermediate inter-connecting network; interconnection may be needed in these cases.

When multiple technologies are employed in the solution, a clear demarcation should be preserved at the inter-connecting points. The problems encountered in one domain SHOULD NOT impact other domains.

From an IP VPN point of view: An IP VPN vPE that implements [RFC4364] is a component of the IP VPN network only. An IP VPN VRF on a physical PE or vPE contains IP routes only, including routes learnt over the locally attached network.

The IP VPN vPE should ideally be located as close to the "customer" edge devices as possible. When this is not possible, simple existing

"IP VPN CE connectivity" mechanisms should be used, such as static, or direct VM attachments such as described in the vCE [I-D.fang-l3vpn-virtual-ce] option below.

Consider the following scenarios when BGP MPLS VPN technology is considered as whole or partial deployment:

Scenario 1: All VPN sites (CEs/VMs) support IP connectivity. The most suited BGP solution is to use IP VPNs [RFC4364] for all sites with PE and/or vPE solutions.

Scenario 2: Legacy Layer 2 connectivity must be supported in certain sites/CEs/VMs, and the rest of the sites/CEs/VMs need only Layer 3 connectivity.

One can consider using a combined vPE and vCE [I-D.fang-l3vpn-virtual-ce] solution to solve the problem. Use IP VPN for all sites with IP connectivity, and a physical or virtual CE (vCE, may reside on the end device) to aggregate the Layer 2 sites which for example, are in a single container in a Data Center. The CE/vCE can be considered as inter-connecting points, where the Layer 2 network is terminated and the corresponding routes for connectivity of the L2 network are inserted into IP VPN VRFs. The Layer 2 aspect is transparent to the L3VPN in this case.

Reducing operation complicity and maintaining the robustness of the solution are the primary reasons for the recommendations.

The interconnection of MPLS VPN in the data center and the MPLS core through ASBR using existing inter-AS options is discussed in detail in [I-D.fang-l3vpn-data-center-interconnect].

7. Management, Control, and Orchestration

7.1 Assumptions

The discussion in this section is based on the following set of assumptions:

- The WAN and the inter-connecting Data Center, MAY be under control of separate administrative domains
- WAN Gateways/ASBRs/PEs are provisioned by existing WAN provisioning systems
- If a single Gateway/ASBR/PE connecting to the WAN on one side, and connecting to the Data Center network on the other side, then this Gateway/ASBR/PE is the demarcation point between the two networks.

- vPEs and VMs are provisioned by Data Center Orchestration systems.
- Managing IP VPNs in the WAN is not within the scope of this document except the inter-connection points.

7.2 Management/Orchestration system interfaces

The Management/Orchestration system CAN be used to communicate with both the DC Gateway/ASBR, and the end devices.

The Management/Orchestration system MUST support standard, programmatic interface for full-duplex, streaming state transfer in and out of the routing system at the Gateway.

The programmatic interface is currently under definition in IETF Interface to Routing Systems (I2RS)) initiative.
[I-D.ietf-i2rs-architecture], and [I-D.ietf-i2rs-problem-statement].

Standard data modeling languages will be defined/identified in I2RS. YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF) [RFC6020] is a promising candidate currently under investigation.

To support remote access between applications running on an end device (e.g., a server) and routers in the network (e.g. the DC Gateway), a standard mechanism is expected to be identified and defined in I2RS to provide the transfer syntax, as defined by a protocol, for communication between the application and the network/routing systems. The protocol(s) SHOULD be lightweight and familiar by the computing communities. Candidate examples include ReSTful web services, JSON [RFC7159], NETCONF [RFC6241], XMPP [RFC6120], and XML. [I-D.ietf-i2rs-architecture].

7.3 Service VM Management

Service VM Management SHOULD be hypervisor agnostic, e.g. On demand service VMs turning-up SHOULD be supported.

7.4 Orchestration and MPLS VPN inter-provisioning

The orchestration system

- 1) MUST support MPLS VPN service activation in virtualized DC.
- 2) MUST support automated cross-provisioning accounting correlation between the WAN MPLS VPN and Data Center for the same tenant.
- 3) MUST support automated cross provisioning state correlation

between WAN MPLS VPN and Data Center for the same tenant

There are two primary approaches for IP VPN provisioning - push and pull, both CAN be used for provisioning/orchestration.

7.4.1 vPE Push model

Push model: push IP VPN provisioning from management/orchestration systems to the IP VPN network elements.

This approach supports service activation and it is commonly used in existing MPLS VPN Enterprise deployments. When extending existing WAN IP VPN solutions into the a Data Center, it MUST support off-line accounting correlation between the WAN MPLS VPN and the cloud/DC MPLS VPN for the tenant. The systems SHOULD be able to bind interface accounting to particular tenant. It MAY requires offline state correlation as well, for example, binding of interface state to tenant.

Provisioning the vPE solution:

1) Provisioning process

- a. The WAN provisioning system periodically provides to the DC orchestration system the VPN tenant and RT context.
- b. DC orchestration system configures vPE on a per request basis

2) Auto state correlation

3) Inter-connection options:

Inter-AS options defined in [RFC4364] may or may not be sufficient for a given inter-connection scenario. BGP IP VPN inter-connection with the Data Center is discussed in [I-D.fang-l3vpn-data-center-interconnect].

This model requires offline accounting correlation

1) Cloud/DC orchestration configures vPE

2) Orchestration initiates WAN IP VPN provisioning; passes connection IDs (e.g., of VLAN/VXLAN) and tenant context to WAN IP VPN provisioning systems.

3) WAN MPLS VPN provisioning system provisions PE VRF and policies as in typical Enterprise IP VPN provisioning processes.

4) Cloud/DC Orchestration system or WAN IP VPN provisioning system

MUST have the knowledge of the connection topology between the DC and WAN, including the particular interfaces on core router and connecting interfaces on the DC PE and/or vPE.

In short, this approach requires off-line accounting correlation and state correlation, and requires per WAN Service Provider integration.

Dynamic BGP sessions between PE/vPE and vCE MAY be used to automate the PE provisioning in the PE-vCE model, that will remove the needs for PE configuration. Caution: This is only under the assumption that the DC provisioning system is trusted and can support dynamic establishment of PE-vCE BGP neighbor relationships, for example, the WAN network and the cloud/DC belong to the same Service Provider.

7.4.2 vPE Pull model

Pull model: pull from network elements to network management/AAA based upon data plane or control plane activity. It supports service activation. This approach is often used in broadband deployments. Dynamic accounting correlation and dynamic state correlation are supported. For example, session based accounting is implicitly includes tenant context state correlation, as well as session-based state that implicitly includes tenant context. Note that the pull model is less common for vPE deployment solutions.

Provisioning process:

- 1) Cloud/DC orchestration configures vPE
- 2) Orchestration primes WAN MPLS VPN provisioning/AAA for new service, passes connection IDs (e.g., VLAN/VXLAN) and tenant context.
- 3) Cloud/DC ASBR detects new VLAN and sends Radius Access-Request (or Diameter Base Protocol request message [RFC6733]).
- 4) Radius Access-Accept (or Diameter Answer) with VRF and other policies

Auto accounting correlation and auto state correlation is supported.

8. Security Considerations

As vPE is an extended BGP/MPLS VPN solution, security threats and defense techniques described in RFC 4111 [RFC4111] generally apply.

When the SDN approach is used, the protocols between the vPE agent and the vPE-C in the controller MUST be mutually authenticated. Given the potentially very large scale and the dynamic nature in the cloud/DC environment, the choice of key management mechanisms need to be further studied.

VMs in the servers can belong to different tenants with different characteristics depending on the application. Classification of the VMs must be done through the orchestration system and appropriate security policies must be applied based on such classification before turning on the services.

9. IANA Considerations

None.

10. Acknowledgments

The authors would like to thank Daniel Voyer for his review and comments.

11. References

11.1 Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001.
- [RFC3209] Awduche, D., et al., "RSVP-TE: Extension to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC4023] Worster, T., Rekhter, Y., and E. Rosen, Ed., "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, March 2005.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [RFC4684] Marques, P., Bonica, R., Fang, L., Martini, L., Raszuk, R., Patel, K., and J. Guichard, "Constrained Route

Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)", RFC 4684, November 2006.

- [RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, October 2007.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.
- [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, March 2011.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.
- [RFC6733] Fajardo, V., Ed., Arkko, J., Loughney, J., and G. Zorn, Ed., "Diameter Base Protocol", RFC 6733, October 2012.

11.2 Informative References

- [RFC4111] Fang, L., Ed., "Security Framework for Provider-Provisioned Virtual Private Networks (PPVPNs)", RFC 4111, July 2005.
- [RFC7159] Bray, T., "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, March 2014.
- [RFC4797] Rekhter, Y., Bonica, R., and E. Rosen, "Use of Provider Edge to Provider Edge (PE-PE) Generic Routing Encapsulation (GRE) or IP in BGP/MPLS IP Virtual Private Networks", RFC 4797, January 2007.
- [I-D.ietf-l3vpn-end-system] Marques, P., Fang, L., Pan, P., Shukla, A., Napierala, M., Bitar, N., "BGP-signaled end-system IP/VPNs", draft-ietf-l3vpn-end-system, work in progress.
- [I-D.rfernando-l3vpn-service-chaining] Fernando, R., Rao, D., Fang, L., Napierala, M., So, N., draft-rfernando-l3vpn-service-chaining, work in progress.
- [I-D.fang-l3vpn-virtual-ce] Fang, L., Evans, J., Ward, D., Fernando, R., Mullooly, J., So, N., Bitar, N., Napierala, M., "BGP

IP VPN Virtual PE", draft-fang-l3vpn-virtual-ce, work in progress.

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

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

[I-D.bitar-i2rs-service-chaining] Bitar, N., Geron, G., Fang, L., Krishnan, R., Leymann, N., Shah, H., Chakrabarti, S., Haddad, W., draft-bitar-i2rs-service-chaining, work in progress.

[I-D.fang-l3vpn-data-center-interconnect] Fang, L., Fernando, R., Rao, D., Boutros, S., "BGP IP VPN Data Center Interconnect", draft-fang-l3vpn-data-center-interconnect, work in progress.

[I-D.ietf-l2vpn-evpn] Sajassi, A., et al., "BGP MPLS Based Ethernet VPN", draft-ietf-l2vpn-evpn, work in progress.

Authors' Addresses

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

David Ward
Cisco
170 W Tasman Dr
San Jose, CA 95134
Email: wardd@cisco.com

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

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

Nabil Bitar
Verizon
40 Sylvan Road
Waltham, MA 02145
Email: nabil.bitar@verizon.com

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

Bruno Rijsman
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
Email: brijsman@juniper.net

Ning So
Vinci Systems
Plano, TX 75082, USA
Email: ning.so@vinci-systems.com

Jim Guichard
Cisco
Boxborough, MA 01719
Email: jguichar@cisco.com

Wen Wang
CenturyLink
2355 Dulles Corner Blvd.
Herndon, VA 20171
Email:Wen.Wang@CenturyLink.com

Manuel Paul
Deutsche Telekom
Winterfeldtstr. 21-27
10781 Berlin, Germany
Email: manuel.paul@telekom.de

Wim Henderichx
Alcatel-Lucent

Email: wim.henderichx@alcatel-lucent.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: August 18, 2014

W. George
Time Warner Cable
R. Shakir
BT
February 14, 2014

IP VPN Scaling Considerations
draft-gs-vpn-scaling-03

Abstract

This document discusses scaling considerations unique to implementation of Layer 3 (IP) Virtual Private Networks, discusses a few best practices, and identifies gaps in the current tools and techniques which are making it more difficult for operators to cost-effectively scale and manage their L3VPN deployments.

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 August 18, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Intention of this Document	3
1.2. Horizontal vs. Vertical Scaling	5
1.3. Developing Requirements for Scaled L3VPN Environments . .	6
2. PE-CE routing protocols	6
2.1. Best Common Practice	7
2.2. Common Problems at Scale Limits	9
3. Multicast	10
3.1. Best Common Practices	10
3.2. Common Problems at Scale Limits	11
4. Network Events	11
4.1. Best Common Practices	11
4.2. Common Problems at Scale Limits	12
5. General Route Scale	13
5.1. Route-reflection and scaling	16
5.2. Best Common Practices	18
5.2.1. Topology-related optimizations	19
5.3. Common problems at scale limits	20
6. Known issues and gaps	21
6.1. PE-CE routing protocols	21
6.2. Multicast	22
6.3. Network Events	22
6.4. General Route Scale	22
6.5. Modeling and Capacity planning	22
6.6. Performance issues	23
6.7. High Availability and Network Resiliency	24
6.8. New methods of horizontal scaling	25
7. To-Do list	25
8. Acknowledgements	26
9. IANA Considerations	26
10. Security Considerations	26
11. Informative References	26
Authors' Addresses	28

1. Introduction

As IP networking has become more ubiquitous and mature, many enterprises have begun migration away from legacy point to point or layer 2 virtual private network (VPN) implementations toward layer 3 VPNs. The VPN implementation as defined by RFC 4364 [RFC4364] enables flexible and robust implementations of IP VPNs. However, in practice, it has become clear that it suffers from significant scaling considerations beyond those discussed in RFC4364. In many

cases, the limits of scale for a given platform are not in sync with the maximum physical and logical interface density supported by the platform, such that a platform may be considered "full" long before the physical slots and ports have all been filled with equipment and connections. This represents an inefficient use of space and power, as well as stranded capital assets, which increase the operator's cost to provide the service as well as the complexity of managing the platform to ensure proper service levels in a wide variety of circumstances. While these scaling considerations are somewhat similar to the scaling concerns experienced in the Global Internet, those are at best a subset of the overall problem, and may not have a great deal of overlap between solutions and best practices. The added complexity and feature set required to support today's enterprise IP networks drives additional scaling considerations for large deployments. A common response to concerns about control plane scale is simply to "throw hardware at the problem" in the form of ever-increasing amounts of memory and CPU resources. In some cases, this may be the only solution, but similarly to the concerns identified in RFC 4984 [RFC4984], there are limits to the growth curve that can be supported and cost-effectively deployed by a VPN provider such that their service remains profitable, and therefore it is necessary to explore the potential for optimization to make the existing resources stretch further.

Generally, router scale can be considered in one of three areas: forwarding capacity, interface density, and control plane capacity. This draft will focus almost exclusively on control plane capacity, because while the others are important considerations for most operators, they are less affected by the details of how L3VPN is implemented either by the router vendor or the operator. Interface density is usually a factor of the forwarding capacity of a given module or slot as well as physical packaging. In this application, interface density is interesting from the perspective of its impact to the control plane - more interfaces means more of all of the different factors that contribute to control plane load, and the operator wants to be able to strike a balance between interface density and control plane capacity such that neither grows out of pace with the other.

1.1. Intention of this Document

This document is intended to provide a discussion of the challenges that network operators face in deploying large-scale L3VPN environments at the time of writing, with two key sets of recommendations. As such, these outcomes can be divided into those that apply to network operators regarding the deployment of particular technologies, and those that apply to network protocol and operating system implementors relating to allowing better

understanding of scaling characteristics in deployments of such equipment.

The best practices defined in this document are intended to allow more optimal scaling of L3VPN networks, whilst minimising the impact on end-customer network behaviour. It is intended that such guidance can be directly utilised by Service Providers to improve the scalability of network elements. However, the guidance in this document should not be viewed as a panacea to the problems of scaling network elements. It is the intention of the authors to document a number of key problems experienced in such environments and provide information to the SP that may result in more optimal deployment of existing technologies to this audience. Additionally, there is a point at which the limits of hardware will be reached, and hence new network elements are required. The key intention of the recommendations provided to Service Providers within this document are intended to allow the resources that exist within existing elements to be utilised in the most efficient manner. Clearly, the optimal point in this balance is that the data-plane and control-plane scale to support similar levels of service termination, so as to result in minimal "over provisioning" of one element.

The scaling considerations presented in this document are intended to provide both network operators and network equipment implementors further guidance around the toolset, and information required to provide accurate means of capacity planning in L3VPN environments. Again, the authors consider that the scaling characteristics, and toolsets required of L3VPN PE equipment diverge somewhat from those required by Internet network equipment. In Internet deployments, relatively standardised interconnects exist across all deployments - typically utilising either static routing, or BGP-4. As such, each connected port comes with a relatively standard overhead in terms of the protocols required. Whilst there is some variance in how "chatty" each customer connection may be, this is balanced by the fact that the whole Internet routing table is typically held on such edge equipment (and hence individual customer's instability tends to be relatively small when compared to the instability of the Internet DFZ). In addition, since such instability is limited to relatively few impacts to a node (interface or BGP session flapping, and BGP UPDATE messages) routers can be optimised to cope with such instability. Counter to this, the L3VPN environment does not have a standardised connectivity model, and typically connects to much less controlled environments. Further details of this are provided within later sections of this document. The result of this difference is that 'headline' scaling figures presented for particular equipment tends to be of limited utility to a network operator. The recommendations within this document outline some of the considerations that must be made in considering the scaling of such

elements, and provide guidance as to the missing inputs and tools that are required to provide information around the capacity of such elements.

1.2. Horizontal vs. Vertical Scaling

Within this document, two forms of 'scaling' are referred to - the "throw hardware at the problem" approach outlined previously involves deploying additional network elements in order to provide further network capacity. Throughout this document, this approach is referred to a horizontal scaling - insofar as it requires parallel deployment of numerous similar elements and balancing the load across the combined capacity of all of the elements. The approach of increasing the capacity of an individual node through allowing the control plane capacity to support the maximum forwarding plane capacity (be it data forwarded, or available ports) is referred to as vertical scaling. It is obvious that at some point the approach of horizontal scaling of elements is required - due to either exhausting available port capacities, or available forwarding plane - however, there are a number of motivations for delaying such provisioning, some of which relate directly to the characteristics of L3VPN environments.

Since a significant proportion of the customers who purchase L3VPN services are Enterprise customers, typically the service is utilised as a WAN for their inter-location connectivity. Clearly, as such customer base tends to be distributed based on differing factors, this implies that such customers connect in numerous geographical locations. The requirement to support service in these locations therefore results in a requirement for the service provider network architecture to support geographically distributed access into such services. A balance must be struck between the extent to which access networks are utilised to backhaul traffic to the service layer, and the geographical distribution of the service layer itself. Both scale and performance characteristics of such networks tend to result in more geographical distribution of service layer elements than in Internet deployments. This distribution results in two particular changes - primarily that the idea of a "point-of-presence" must be reconsidered - where an assumption in Internet environments may be that there are separated core and access elements within a single location, within a distributed L3VPN environment, a point of presence may be a single PE device. The result of these small scale points of presence is that numerous core and edge functions must be collapsed onto a single device. For this reason, the approach of adding additional devices to the network may have an impact on a further subset of devices within the network (particularly due to any mesh-based protocols that are deployed), and hence result in a change in the scaling characteristics of these devices. In this case, there

is further motivation to avoid large numbers of devices in the network where possible. Further to this, the smaller PoP profile may result in physical constraints around the deployment of additional network elements, particularly due to the availability of power and physical space to deploy such elements.

1.3. Developing Requirements for Scaled L3VPN Environments

Whilst the collected scaling considerations outlined in this document are based on the author's collective experience within various Service Provider networks, and discussions with operators of similar networks, it should be noted that the problems outlined in this document are not static. With the growth in the use of IP as the underlying transport of many services, the demand for L3VPN environments has grown. As such, this has meant that various technologies are being considered to allow growth of these networks at a lower cost point to a wider footprint than was previously required. A network operator must therefore consider the extent to which the service layer must be built - both to meet economic and technical requirements. With newer aggregation methods, the service layer edge (and hence the L3VPN PE) acquires responsibility for inter-working between newer dynamic aggregation technologies, and the existing IP network. As such, these edge functionalities result in further requirements for loading onto these network elements.

*** Author's note: Do we want to put anything about NNI for footprint extension here? Datacenter edge - perhaps Ning's problem around the L3VPN edge in his datacentres? ***

2. PE-CE routing protocols

One of the things that makes IP VPNs so flexible and robust is their ability to participate in the encapsulated network's routing protocols, where the customer edge (CE) router has a direct neighbor relationship with its upstream provider edge (PE) router in order to exchange routing information about the Virtual Route Forwarding (VRF) instance that represents the VPN. In many cases, this is managed through a combination of static routes and BGP neighbors, but IGPs such as OSPF RFC 4577 [RFC4577] are often supported, because it enables a more complete integration into an existing enterprise network design and topology. In some single-vendor implementations, carriers sometimes support proprietary routing protocols such as EIGRP [EIGRP]. IGPs may also be chosen due to a belief that they will respond more rapidly during a failure than BGP will. In reality, this may not be true due to the fact that VRF routing information is still carried in MP-BGP from PE to PE, and the PE-CE routing protocol's characteristics are only locally significant. In

fact, the increased overhead may lead to slower convergence times than a more standard BGP implementation.

IGPs often translate to a significant increase in overhead due to their inherent characteristics as link-state routing protocols requiring full topology databases and flooding of updates to all participants, and the fact that they invoke additional processes on the router when compared to simply using BGP (which is already going to be running on a router using MP-BGP for VPNs). While a router may be able to scale almost effortlessly with a few thousand routes in a single IGP plus hundreds of thousands of routes and many neighbors in BGP, it may be quickly challenged if it is also required to run multiple instances of an IGP each with a certain number of routes that must be moved into MP-BGP to be passed to the rest of the VPN infrastructure. The advent of support for IPv6 within a VPN (6VPE) [RFC4659] has the potential to make this problem worse, especially in the case of OSPF, where it now requires both OSPFv2 [RFC4577] and v3 [RFC6565] to run as separate instances for the two address families, or use of multiple instances of OSPFv3 to support multiple address families as documented in RFC5838 [RFC5838].

Another consideration in PE-CE routing protocols is the timers used for each session. These will be discussed in greater detail in the best practices section.

2.1. Best Common Practice

Ultimately, the decision as to which PE-CE routing protocols to support is a business decision much more often than it is a technical one, because there are few use cases where something other than BGP and static routing as PE-CE routing protocols is a technical requirement. If a provider chooses to support additional protocols, especially IGPs, they should consider the effects that these have on the overall scaling profile of the PE routers and the network as a whole when determining if and to what extent they will support other protocols.

Often, those designing VPN solutions attempt to use extremely aggressive routing protocol timer and keepalive values as a means of rapid failure detection and reconvergence. This tends to make PE-CE routing protocols more fragile and increase the load on the PE router with questionable benefit. This is especially common in scenarios where the network designer is attempting to replicate native IGP-like failure detection and reroute capabilities using BGP. In order to avoid this, the preferred values should be set to something that is appropriate for large-scale implementations. Further, because timer and keepalive values are often negotiated based on the more aggressive neighbor, it is a good idea to set a minimum acceptable

value, so that instead of being forced to support negotiated timer values that are too aggressive for the scale that a given PE router is expected to support, the neighbor session will simply stay down until the remote end timers are reconfigured to a more acceptable value. This acts as a safety valve against abuse that can destabilize a router used by multiple customers. Because aggressive timers may be unavoidable in certain situations, it may be advisable to track the number of sessions which are provisioned with aggressive timers vs how many are using more conservative timers on a per-router basis, so that effort can be made to balance aggressive and conservative timers on each router. This will help to prevent "hot-spots" where given a similar port and VRF density, some routers have significantly higher CPU usage in steady-state than others.

It is important to realize that while use of aggressive routing protocol timers is not a scalable way to do fast failure detection, fast failure detection is still a requirement for many customers. Because this is becoming such a table-stakes requirement, the provider must consider other alternatives such as Bidirectional Forwarding Detection ([RFC5880]), Ethernet OAM 802.lag [IEEE802.1], ITU-T &.1731 [Y.1731] LACP 802.3ad [IEEE802.3] and the like. These extensions often come with their own scaling considerations, but more and more they are implemented in a distributed fashion so that instead of affecting the main router CPU like a routing protocol might, they offload that processing to the linecard CPU, and therefore can support more aggressive scale. The general philosophy is that these lower-layer detection mechanisms should serve as the primary detection and failure point, with the upper layer routing protocols only serving as a backstop if the failure is not detected by the lower level protocols for some period of time.

Another important consideration is that there is not likely to be a "one-size-fits-all" solution when it comes to setting timers and policies around PE-CE routing protocols. At a minimum, a distinction should be made between sites that have only a single upstream connection and those that have two or more diverse connections to the network. Further distinction can be made based on the importance of the site, whether it is a hub site or an end site. These can all be used to determine the aggressiveness that is appropriate for the timers and perhaps even which routing protocols are appropriate. For example, an end site with a single upstream connection likely does not need very aggressive timers and may be able to get by using only static routing, while a hub site with multiple connections and a need for rapid restoration and reaction to any routing changes may need BGP along with aggressive lower-layer timers for fault detection.

2.2. Common Problems at Scale Limits

Two common problems when working on a heavily-loaded system:

CPU cycle constraints, even before the system reaches the point of scheduler thrashing often lead to one or more routing protocol neighbor hello drops. If several consecutive drops occur, the remote neighbor may declare the session dead, which triggers a restart of the connection and a resync of the routing data. Because this connection initialization requires dedicated CPU cycles to generate, receive, acknowledge, and process the updates, it increases the CPU utilization further, which may trigger additional hello failures and neighbor resets, resulting in a snowball effect where a relatively minor event rapidly becomes a major one due to interactions between multiple scaling limitations. This problem is made worse by extremely aggressive timer values, because they raise the baseline CPU load with more frequent hellos and responses, and are more sensitive to drops caused by increased CPU load. Further, because failures brought on by loss of hello packets are unlikely to invoke any graceful restart [RFC4781] machinery that the system may support, it is unlikely that the session reset will be able to take advantage of optimizations like only synching the changes that occurred while the session was dead, thus increasing the outage time and the CPU cycles to get things back into sync.

Another potential issue during times of high-CPU operation is related to process prioritization. This is applicable in different ways for both multithreaded and interrupt-driven OS architectures. In each case, the scheduling algorithm that the router uses to prioritize different CPU cycle work items and manage the timeslices individual tasks are given to complete may require significant tuning and prioritization in order to ensure the desired behavior during high CPU usage. Improperly tuned or prioritized processes may significantly delay completion of routing table/update processing such that it may take an excessive amount of time for the routing table to converge properly. This issue is further exacerbated if the VRF instance has a large amount of routes, or is prone to frequent event-driven route churn. In some cases, the routing table in a given VRF may never fully converge, leading to routing loops, traffic loss, inconsistent latency, and a generally adverse customer experience.

These items also can have a cascade effect on other routers in the system if they also participate in a given VRF that is being affected by this type of scaling issue. Not only is the local PE router affected, but any upstream Route reflectors, as well as other PEs, and even CEs participating in this VRF will see increased CPU cycles

in order to receive and process the increased flow of updates driven by the local churn.

specific items related to different PE-CE protocols?

3. Multicast

Multicast support within a VPN [RFC6513] has become an increasingly popular feature, but comes with its own scaling considerations. Depending on the application, the frequency at which multicast state changes within a given VPN (e.g. PIM joins and prunes) will contribute to the CPU load on the router, and any instability in the network can potentially increase these as remote sites flap. In extreme cases, PIM neighborships can be lost during events, disrupting the flow of multicast traffic.

It should be noted that, in some cases, dynamic action is required by a PE device to support the transition of flooding of multicast data from a non-optimal distribution tree (the default MDT in [RFC6037], or the I-PMSI) onto a more optimal one (a data MDT or S-PMSI). Where such a transition is required, consideration is required of the nature of the traffic sourced by an end user of the L3VPN service. The net result of this consideration is that it becomes increasingly difficult to reliably gauge the scaling impact of specific end-site deployments. Additional scaling considerations around Multicast in a VPN are related to the size and number of multicast streams. While this is a consideration whenever Multicast is used even outside of a VPN because of the bandwidth utilization it may generate in the core, the additional overhead of implementing multicast within a VPN makes this a more significant consideration in this case. Related to the previous consideration is the stream fanout - the amount of P and PE router paths in the network that could potentially carry a given multicast stream based on the number of PEs that are configured with a given Multicast-enabled VRF, and the number that actually do carry the stream based on actual receivers joining the stream behind that PE.

*** This section is quite weak. We're looking for contributors who can assist in fleshing this out ***

3.1. Best Common Practices

Multicast BCPs???

3.2. Common Problems at Scale Limits

Multicast tree interruptions

PIM neighbor adjacency drops

4. Network Events

Network events are an important scaling consideration because they can have wide-ranging impacts far beyond the individual VRF or even PE router that experiences the event. At high scale, a seemingly innocuous event on one router or VRF can trigger secondary impacts and outages on remote routers elsewhere in the network. Correlating these events for root cause analysis can be challenging by itself, and trying to characterize the impacts as they relate to scale in a way that informs the provider's decisions is even more difficult. Different types of Network Events that can contribute are: Interface flaps, hardware and software outages (both planned and unplanned), externally driven route-churn events (such as those that originate on an NNI partner's network) and configuration changes.

4.1. Best Common Practices

While this document suggests that lower layer failure detection protocols like BFD and Ethernet OAM be more aggressive so that routing protocol timers can be more conservative, it is still important to remember that this can generate false positives or excessive churn that will cascade into a scaling problem at other parts of the system, so the timers should not automatically be configured to their minimum supported values. Rather, each application may be slightly different, and the timers should only be set as aggressively as necessary to ensure acceptable performance of the applications in question. It may be appropriate to set limits (e.g. in provisioning logic/rules) as to the number of interfaces per router and per VRF that can use aggressive, moderate, and conservative interface timers.

Even with timers set as conservatively as the application will allow, churn is unavoidable. For this reason, it is also a good idea to use interface-level dampening such as hold-down timers or event dampening in order to ensure that interfaces that flap too rapidly will not telegraph that churn into the upper-layer routing protocols any more than necessary. BGP Peer Oscillation Dampening (DampPeerOscillations, RFC4271 [RFC4271]) may also help to reduce intermittent outage-based churn while leaving the interface itself unaffected. All of these dampening measures help to ensure that problems are localized to a single PE or even a single interface,

rather than causing instability and routing churn throughout the VRF and the provider network.

In addition to interface dampening, it may be advisable to consider implementing some manner of route flap dampening route flap dampening (RFD) [I-D.ietf-idr-rfd-usable] to assist in reducing the impact that route churn may have on the SP's network infrastructure. This is currently fairly uncommon within VPN environments, and is not without controversy. While it may help with scaling, it also requires each PE to maintain more state to store and compute the per-prefix penalty values, which may reduce the benefits gained by implementing RFD. Further, customers typically expect a fair amount of transparency in the provider's participation in their routing instances. Many providers and customers view a VPN or VRF as a part of the customer's internal network and therefore compartmentalized so that the customer can only affect their own routing if they have a problem with excessive route flaps. Further, if routes are dampened it requires intervention from the SP to clear the dampening, which can potentially add to the outage time that a customer experiences once the issue that triggered the dampening is resolved. Implementing RFD may even drive the need for a customer-accessible looking glass, which is far more complex in the VPN space owing to the requirement to prevent one customer from looking at another's VRF routes on a common platform.

4.2. Common Problems at Scale Limits

Network events are both a cause and a symptom of a system running at or near its scaling limits. As noted above, event-driven routing table churn or routing protocol interactions can significantly drive up CPU usage on the locally connected PE as well as on other PEs and CEs participating in the VRF. If routes are constantly changing due to a preferred path repeatedly being added and removed, latency and jitter numbers can be affected in a way that adversely effects applications sensitive to this sort of change. Network events can also be triggered by routers with high CPU, because similarly to systems which may have aggressive routing protocol timers for enhanced failure detection, systems with centralized CPU-based implementations for lower-layer protocols (such as HDLC [ISO13239] PPP [RFC1661], LACP, BFD/EOAM) may start losing keepalives and declaring outages that result in physical interfaces being torn down and restored. Again, implementations that choose timer and multiplier values or numbers of sessions at or near the maximum rated scaling for the device put the operator in a position where there is very little headroom to deal with an event that momentarily spikes CPU usage, meaning that the likelihood of a cascade failure dramatically increases.

As above, these network events may be something that occurs elsewhere in the network, and may trigger a failure on a completely different PE or CE router. The danger with this is that it is extremely difficult to troubleshoot and correlate root causes when the outage observed isn't caused by an event on the same router. Failures become increasingly non-deterministic and difficult for operators to manage and address.

5. General Route Scale

PE routers in a carrier network can have many different implementation scenarios. Some carriers implement a dedicated PE router that is only responsible for carrying VPN routes and therefore may only carry IGP routes in its global routing table, rather than a full internet routing table. Others use combined edge routers that carry full routes plus a complement of customer VPN routes, and some even place the full internet routing table into one or more VRF instances. The issue here is that the weight of all of these routes and paths must be combined when considering the maximum scale of the router, both in terms of memory footprint and in terms of convergence times. The addition of an 8-byte RD appended to the IP address to ensure uniqueness means that each VPN prefix takes up incrementally more physical space in memory than an equivalent non-VPN route. Further, the greater number of Address-families running simultaneously on the same router, the more sensitive it will be to event-induced churn since each address-family (and VRF) often has its own independent computation/SPF run. The addition of IPv6 support within both the global routing table and within a VPN adds yet another source for routing table bloat. A PE router can be running a combination of any of the following address-families:

- o Global IPv4 unicast
- o Global IPv4 multicast
- o VPN IPv4 unicast
- o VPN IPv4 multicast
- o Global IPv6 unicast
- o Global IPv6 multicast
- o VPN IPv6 unicast
- o VPN IPv6 multicast

Even PE routers that do not carry the full internet routing table are still required to carry a minimal number of IGP routes, LDP information, and some amount of TE tunnel state, adding to the items competing for scale. On high-scale PE routers, the VPN routing tables are often as large as or larger than the equivalent global routing table in both number of routes and number of paths. This is at least partially due to the fact that there are no constraints on the customer addressing plan within a VPN other than they cannot conflict within a given VRF, or with any extranet with which the VRF interconnects. As such, they may not necessarily adhere to any best practices to control the deaggregation of the routing table such as hierarchical addressing, aggregation and summarization of announcements, and minimum prefix lengths. It's also quite likely that connected interfaces will be redistributed, and little or no route filtering may take place. Most PE routers use the absence of a given VRF instance (or RD/RT filtering) to limit the number of routes that they must actually carry, but this is sometimes of limited utility for a couple of reasons. First, it leads to an inconsistent routing table footprint from one PE router to the next, and it can change with every new customer turned up on the router. These lead to non-deterministic performance and scale from PE to PE and from customer to customer. In other words, PE1 may be fine from a scale perspective, while PE2, which has the same number of occupied ports has significant scaling problems on account of which VRFs are present /absent. Then, PE1 may find itself suddenly having the same scaling concerns because a new customer was provisioned with a large or high-churn VRF that was previously not present on the router. Second, many customer VPNs are so large and have such stringent diversity requirements that they have a presence on nearly every PE router in a provider's network, meaning that one cannot rely heavily on statistical distribution to reduce the percentage of VRFs that must be installed on a specific PE router. In addition, customers may request the use of BGP multipath for faster failover or better load balancing, which has the net effect of installing more active routes into the table, rather than simply selecting the single best path. The scaling considerations for enabling BGP Multipath are not unique to L3VPN, but they are more pertinent here because SPs are less likely to be willing to enable MP for standard internet traffic, while they will do it for L3VPN. The application as an enterprise network instead of internet connectivity drives a different set of expectations about the performance of the network, design tradeoffs that must be made to meet the SP's requirements, etc. In many cases, L3VPNs are replacing old point-to-point networks or L2VPNs using legacy Frame Relay, ATM, or L2TPv3. Customers often don't want to make major architectural changes to their routing, and therefore expect the SP to do the same things that they were doing between their routers before, including things like multipath.

In addition to such intended behaviour, within many L3VPN networks, a balance must be struck between complexity in OSS such as provisioning and inventory systems, and complexity in network deployments. One such example of this is the assignment of route distinguisher (RD) attributes. Where it may be possible to assign a single RD per L3VPN instance, and hence achieve some level of route aggregation for multi-homed CE routes on BGP speakers within the solution, this has some consequences for both convergence in the VPN (due to BGP convergence being relied upon) and in its potential to exacerbate geographic distance between PE and Route-reflector and is therefore undesirable in some circumstances. In order to avoid this, multiple RDs are then required, which requires OSS and inventory support to control the namespace. As such, due to this requirement, often each VRF instance is deployed with a specific RD - which, whilst achieving the desired convergence effect, places load on all BGP control-plane elements of the provider network.

Total supportable route scale on a given PE router will be driven by multiple different variables, which have a roughly inverse relationship to one another: Number of VRFs per router, number of routes per VRF, number of neighbors per VRF. For example, a router can support a low number of VRFs per router if each VRF has a large number of routes per VRF and/or a large number of neighbors per VRF. Conversely, a router can support a relatively high number of VRFs if each VRF is kept to a much lower number of routes per VRF, and/or lower numbers of neighbors per VRF. This provides a baseline that then must be reduced based on the expected level of event-driven churn, the type of protocol chosen, etc. In short, this is a difficult problem from a modeling and capacity planning perspective.

It is fairly common for the contract or Service Level Agreement between SP and customer to include a maximum limit as to how many routes can be carried in a VRF. At its most basic, this maximum can be used as a method to estimate the number of VRFs that can be present on a PE given its scaling limitations. However, there is a wide gulf between a contractual limitation of no more than N routes per VRF with a corresponding configured limit and the fact that many customers will not carry nearly that many routes. This leads to the potential for significant stranded capacity. Therefore the provider needs a way to have different tiers of "maximum routes allowed" so that the capacity management can be done in such a way as to enable better loading of PE routers to take this relationship into account (e.g. populating a PE with a combination of high-scale and low-scale VRFs). The alternative to this method would be to assume a standard maximum routes per VRF, and then similarly to the way that carriers use statistical multiplexing and oversubscription to assume that not all customers will have their pipes full of bandwidth at the same time, make some assumptions about control plane capacity. This may

come in the form of an average that is calculated based on the actual size of the routes in each VRF. This has many challenges. Among them- Should it be calculated per-PE? Network-wide? What happens when there are too many VRFs that exceed the average on a given PE? How does one add control plane capacity to a "full" router? This may be a manageable model in a network with a robust and flexible provisioning system, such that high-scale VRFs can be moved between PE routers to balance the load, but each of these moves likely represents an outage for the customer and the potential for other errors to creep in, and is not likely to be attractive due to the operational costs of managing the network. In other words, it doesn't scale, but for a completely different reason. Further, this VRF route limit may or may not be a physically enforced value. Some PEs have an additional configuration knob per VRF that places a hard limit on the number of routes the VRF will accept. This works well as a last-chance safety valve to protect the PE and the network in the case where there are misconfigurations in the VRF that cause a sudden and significant increase in the number of routes, but can create inconsistencies in the VRF's routing table if there is a periodic or intermittent increase in the routes that causes the maximum to be periodically exceeded. Unlike something like a BGP maximum prefix limit, which shuts down the BGP neighbor when a threshold is exceeded, there is no direct feedback to the peers that the VRF route limit threshold is exceeded, and different implementations handle this in different ways in terms of how they drop or buffer routes, and how they resynch once the routes are below the threshold again. It may be appropriate to identify a common way for implementations to handle this limit, perhaps triggering one or more PE-CE peering sessions to drop, etc. so that this is a more useful tool to protect the PE from increases that would cause it to have scaling problems.

5.1. Route-reflection and scaling

Most of this document focuses on scaling at the PE router, but a discussion of route scaling would not be complete without at least a cursory mention of route-reflection [RFC4456]. While using route-reflectors to eliminate the need for a full mesh of your PE routers is a common optimization, there are many different deployment models as far as whether dedicated route-reflectors are deployed vs. running an existing PE or P router as a route-reflector, how many are deployed and where, the method for ensuring diversity and redundancy, and even whether a router is used vs. a commodity PC running some sort of routing daemon. From a scaling perspective, there are several considerations that are unique to the route-reflector design that will be discussed here.

Starting with the route-reflector itself, these devices are often experiencing a worst-case scenario when it comes to storing entries in the RIB, exposure to route-churn, etc. This is because they are not capable of filtering the routes from VRFs not locally configured on themselves, and they must carry all of the routes for all of the VRFs in the ASN. This requires significant amounts of CPU and memory to store and manage these updates, and an underpowered route-reflector can quickly cause widespread convergence problems if it is unable to keep up with the load of receiving, processing, and propagating these updates. Beyond CPU and memory, it may also be necessary to know how the router manages its FIB when running as a route-reflector. A route-reflector is almost 100% control-plane, but if it tries to install all of the routes that it has in its RIB into the FIB, it may require very high-scale (and therefore costly) forwarding hardware to manage the large FIB. It may be useful to select a device that is capable of optimizing for this control-plane only mode and suppressing unnecessary routes from its FIB to reduce the overhead. This is why some providers choose to use commodity PCs, which are well-suited for high-scale, processor and memory-intensive control plane work, and can easily and cost-effectively be horizontally scaled. The main consideration with using a PC instead of a router for route-reflection is that there may be implementation differences that lead to incompatibilities in terms of supported features, and there may be a different model in terms of how high-scale applications are managed, or even what bugs are exposed at maximum scale, all of which will require significant additional testing.

Route-reflector placement is another important consideration. Because route-reflectors are control-plane devices, and the scale requirements for them are high enough that they can be expensive, the tendency might be to deploy two large geographically-diverse and horizontally scaled sets of them in order to provide an acceptable amount of diversity while deploying the fewest possible devices. However, this leads to potential problems with the geographic distance between the PE and the route-reflector leading to geographic "routing artifacts". (Geographic routing artifacts in this case is referring to the phenomenon where the PE and the route-reflector are significantly distant from one another in the network, and the route-reflector chooses one or more best paths based on its view of the IGP, and then reflects those to its neighbors, even though there may be a better path at a given PE based on its location in the network and its view of IGP. Also, propagation delay and the latency it induces for updates and convergence may be a factor.) Use of a small number of route-reflectors network-wide can also result in scaling problems based on the number of BGP sessions a given route-reflector must maintain. Both of these items point to a larger deployment of smaller, more geographically diverse route-reflectors throughout the

network, so that a given route-reflector is maintaining fewer BGP sessions with PE routers, has an IGP view of the network that is closer to that of the PEs it peers with, and can rapidly propagate local updates to the surrounding PEs.

The number of route-reflectors peering with each PE is a scaling consideration as well. While a minimum of two discrete route-reflector BGP sessions is the minimum to ensure proper redundancy, adding additional route-reflectors requires each PE to carry the additional state of those sessions, adding significant overhead with questionable value.

Related to route-reflector placement and the number of PE to route-reflector peering sessions is the use of cluster-IDs within a set of route-reflectors. Cluster-IDs can be effectively used to reduce the amount of duplicate route updates propagated between route-reflectors, thus reducing some of the same state and churn impact that is so critical in high-scale implementations. However, it can have unintended side effects. In order to prevent inconsistency in the routing table, a PE MUST peer with all of the route-reflectors in a given cluster. As a result, depending on how route-reflectors are spread out throughout the network and clustered together, it may create the need for a PE to either peer with multiple clusters, or to peer with one or more route-reflectors that are not optimal in terms of geographic placement in relation to the PE. For example, if each cluster has two route-reflectors for redundancy, and there are three regional clusters (East, Central, West), PEs that sit in the overlap area between two cluster "regions" may have to peer with one or more route-reflectors that are farther away, lest they have to now peer with four route-reflectors in order to peer with the two closest to them.

5.2. Best Common Practices

A number of things can be done to improve the general route scaling. Most BGP sessions can be configured with a similar set of protections as they would be if they were global Internet eBGP sessions, such as maximum prefix limits, inbound and outbound prefix filtering, etc. Prefix filtering is less common within VPNs because it is treated more like iBGP, where filtering is typically not recommended (***reference?***), or as noted above, it's part of the customer's network and therefore not the SP's business/problem to do filtering in an application that can only break that customer's network. What is often more important in the case of individual VRFs is to configure an acceptable maximum number of routes that the VRF is permitted to carry. This allows the SP to control their exposure to sudden increases in the memory footprint of the routing table, especially if a misconfiguration on the CE side leads to significant

amounts of route leakage, such as to suddenly leak a significant amount of the Global Internet Routing Table into their VRF. However, it can also be used to enforce the assumptions on number of routes per VRF that the SP has used to determine what the other max scaling values such as number of VRFs per router, number of sessions per router, etc.

As noted above, the number of VRFs per router, number of routes per VRF, and number of sessions per router and per VRF are all inter-related values in the way that they contribute to overall router scale. The more of this information is known in advance based on the design of the customer's network, the more it can be used as input to the provisioning system to determine the best available PE router on which to terminate the connections for consistent loading. Since these values are usually estimates, and considerations like diverse router terminations may drive a specific choice, this is not by any means fool-proof, but is a valuable optimization to improve the density of customers on a given router and maximize the return on investment for the capacity deployed. It is worth noting, however, that many SP VPN networks have a different geographic spread than do their Internet service counterparts, where there will be more POPs with fewer routers, as it is important to provide more local handoffs to customers. This may limit the SP's flexibility in terms of homing locations and router choices, and thus may be of limited value when controlling scale impacts on individual PE routers.

*** Authors' note: Should we discuss incremental SPF, next-hop tracking, SPF timer tuning (By protocol and AF), prefix prioritization, etc? All of these are generally thought of as convergence optimizations, and may be applicable here as a way to both reduce the CPU load and ensure that behavior is more deterministic, but I'm not sure how much depth we want to get into here, especially since some are vendor-specific or FIB-specific optimizations... ***

5.2.1. Topology-related optimizations

As has been discussed above, the topology of a given VPN and its placement on the available PE routers can be a significant contributing factor to the impacts of that VPN on the scaling limits of a given PE. For example, a hub and spoke arrangement allows for some amount of aggregation and route summarization to be used, but there are limitations to its effectiveness at minimizing routing table growth since this is typically implemented by the end customer, and is dependent on how hierarchical their topology and IP addressing plan is. While there are plenty of other good reasons to use a hub and spoke design, including security (traffic separation) between spoke sites, etc., generally, a customer does not have much incentive

to expend the time and effort to maintain a proper hierarchy or deal with the added complexity of a hub and spoke design if the only benefit is to improve route scaling. A possible solution for some full-mesh topologies is to use Virtual Hub-and-Spoke in BGP/MPLS VPNs [RFC7024]. From the abstract:

"With BGP/MPLS VPNs, any-to-any connectivity among sites of a given Virtual Private Network would require each Provider Edge router that has one or more of these sites connected to it to hold all the routes of that Virtual Private Network. The approach described in this document allows to reduce the number of Provider Edge routers that have to maintain all these routes by requiring only a subset of these routers to maintain all these routes."

The value of this approach is that it is much less dependent on the individual customer to implement a hierarchy in order to conserve routing table entries. The potential downside to this approach is that it requires additional provisioning and troubleshooting complexity due to the way that routes are/are not imported, the use of default/summary routes, etc. This approach also potentially exacerbates the problem discussed above where PE's are inconsistently loaded (in terms of total number of routes) from one PE to the next and the potential provisioning difficulty that comes from a desire to find and use as much spare control plane capacity as possible without overloading a given PE.

5.3. Common problems at scale limits

As mentioned above, systems that are carrying a large number of VRFs and/or VRFs with large numbers of routes tend to be more sensitive during events due to the increased amount of periodic and event-driven processing that must be done to complete a walk of the routing table to process updates. While optimization techniques may reduce the overhead of (re)programming the FIB after an update, there are less tricks to be employed in managing the RIB, and they are often vendor-specific, which leads to a lowest-common-denominator threshold in multivendor environments.

In addition to CPU constraints, it's common for route memory footprint to be a consideration if there are large numbers of VRFs with large numbers of routes. Similarly to the way that high scale reduces the cushion of available CPU resources to absorb temporary peaks, as memory use reaches its high threshold, allocation of the remaining memory becomes less efficient and more fragmented, such that memory allocations may begin to fail well before the available memory is actually exhausted. Depending on the specific implementation, the "largest free" may be more important than the "total free" and it may be difficult or impossible to coalesce the

free memory to reduce fragmentation to an acceptable level. As with other scaling problems, a failure of this type has the nasty habit of causing a cascade of problems. Depending on how robust the system is at recovering from memory allocation failures, it may trigger restarts of critical routing processes or even the entire system. These may or may not be graceful and hitless, and even if they are locally a fairly low impact, these may trigger events on other routers due to the ripple effect of the network event itself. It is also worth noting that there are hardware and software limits to how much memory a given system can use - if the router in question does not use a 64-bit OS, then it is unable to address more than 4GB of RAM, for example. This may make an otherwise robust system incapable of scaling to the necessary level, and make memory usage an even more significant consideration.

6. Known issues and gaps

6.1. PE-CE routing protocols

While support for route flap dampening in BGP as a PE-CE routing protocol is equivalent to its support in non-VPN applications, the addition of IGP routing protocols such as OSPF creates a new problem, in that there is not really a way to manage route dampening, either by configuring it within the context of the IGP itself, or by configuring it in the translation point where the IGP's routing information is moved into the MP-BGP control plane infrastructure to be exchanged between participating PEs across the VPN network. This means that in the case where IGPs are used, which is often more CPU-intensive and performance-conscious to start with, the route flaps associated with an unstable network will make a bad problem even worse. It may be advisable for the IETF to document updates to standards managing use of IGPs as PE-CE routing protocols to explicitly define the use of RFD in this application.

There are also not clear guidelines based on testing and real-world experience for recommended timer values or appropriate use cases for an IGP vs BGP as a PE-CE routing protocol. In other words, rather than enterprises simply defaulting to whatever IGP is already in use or they are most comfortable with, there may be certain cases where use of an IGP is recommended, and those where it is not. Guidance in this area may be very useful to both the SPs supporting these networks and the engineers designing the corporate networks that make use of them.

6.2. Multicast

Issues in multicast VPN scale?

6.3. Network Events

Guidance on interface event dampening values (research and testing), correlation tools to help determine root cause in a cascade failure,

6.4. General Route Scale

Route flap dampening may potentially be a best practice, but it has a number of shortcomings. First, there is no systematic way for end customers to view and clear dampening without some sort of advanced-functionality looking glass that allows them to view only the routes in their authorized VRFs. Also, allowing customers to make unattended clears of dampened routes may defeat the purpose of having dampening enabled at all, since customers may clear the dampening without addressing the underlying cause of the problem. In addition, as noted in [I-D.ietf-idr-rfd-usable] and [I-D.shishio-grow-isp-rfd-implement-survey], Route flap Dampening is not widely used even within the Global Internet routing table, and its values probably need to be tweaked. Due to the differences in the characteristics of VPN routes compared with the global routing table, additional study and recommendations as to appropriate RFD values within a VPN are likely required. Additionally, it is not possible to configure RFD on IGP, either natively within the PE-CE routing protocol or upstream where the learned routes are carried in MP-BGP. This means that in some cases, there is no way to insulate the SP network from the adverse impacts of rapid route churn.

6.5. Modeling and Capacity planning

There is a significant lack of multidimensional scale guidance and modeling for capacity planning and troubleshooting large-scale VPN deployments. This has a number of contributing factors. First, behavior at scale becomes increasingly non-deterministic the more variables you're working with simultaneously, so this is classically a difficult problem to model. Even worse, it's difficult to account in a model for latent design/implementation flaws: things that work well enough at moderate scale, but are not efficient enough for high scale, or suffer some sort of secondary impact due to dependencies, race conditions, etc. These problems are often only found through extensive testing or even escape into production. Second, it is difficult to characterize an "average" implementation in such a way that it can be tested to failure in multiple permutations to provide a reasonably accurate multidimensional model. Consequently, the guidance available normally takes the form of multiple uni-

dimensional scale thresholds plus some very conservative multi-dimensional thresholds. These conservative recommendations avoid risk to both the vendor and the implementer by catering to the lowest common denominator, but they have the adverse effect of leaving a lot of capacity sitting idle. Some vendors make an effort to characterize their customers' large scale implementations such that they can better replicate real-world conditions, but gathering this information and devising ways to replicate the behavior in a lab is problematic and time-consuming.

This leads to a follow-on issue, which is that there is a lack of instrumentation on critical scaling vectors. Some routers have very limited abilities to provide useful data about critical scaling vectors (routing updates per second, changes in multicast state, sources of internal bottlenecks, etc), either for use in a model or for use as additional capacity monitoring thresholds. While most routers can provide information about CPU usage and memory thresholds, and even which processes are consuming large amounts of resources, it often takes special instrumented versions of the OS to provide a window into what is actually causing some sort of failure at scale. Because these are not routinely monitored, it means that the provider may be blind to one or more early warning signs that the router is nearing its scaling limits and cannot take action to prevent exceeding those limits before it causes customer impacts.

Additionally, even if this information is available, the provisioning systems used by most providers do not currently have the intelligence or visibility to make a decision regarding which PE to provision new customers on to evenly load the available PE routers. The provisioning system is often aware of the available physical or logical port capacity on a given router or site, and uses this as a key input to its port choice for newly provisioned customers. However, these additional capacity and scale vectors are based on real-time statistics from the router (CPU, memory load, etc) and there is no interaction or feedback loop between the provisioning system and these types of real-time router scale stats. As a result, manual intervention is often required to either remove busy routers from the available capacity pool, move spare port capacity from a busy router to a full one, or even to reprovision customers to move them from one device to another to rebalance the load on each router.

6.6. Performance issues

In many ways, it's difficult to define a hard-and-fast scale limit, because each provider and customer have a differing view on what is an acceptable performance envelope both in steady state and during recovery from outages, whether planned or unplanned. In the most extreme sorts of network events, such as a heavily loaded PE router

undergoing a cold restart, the scale considerations may take something like boot and convergence times from what the involved parties consider acceptable and extend them to the point where they significantly prolong the pain that to which an end customer is exposed. They often have the added problem of making it difficult to predict the duration of an outage, because individual customer VRFs may be affected for differing amounts of time based on all of the factors that contribute to scaling and affect convergence. For example, if a customer has one critical route that happens to be among the last to converge, they perceive the outage to be ongoing until that last route converges, even if the entire rest of their network has been functional for a significant amount of time prior to that point.

When dealing with scheduled outages, customers obviously prefer that they never are impacted. Since this is not really possible, they expect the provider to give them very clear and accurate guidance on what the impacts will be, when they will occur, and for what duration, so that they can set expectations for their customers. VPNs are often carrying mission-critical services and data, so any downtime is bad downtime. While a customer may be understanding of a scheduled maintenance with a 15-30 minute traffic interruption while a router reloads, they may be less so if the outage actually stretches for 60-90 minutes while the router runs at 100% CPU trying to deal with this worst-case sort of load or suffers intermittent cascade problems while any remaining cushion is used up dealing with the results of the event. These impacts may be largely invisible to the provider unless they have probes within each VRF or other means to verify that traffic is no longer impacted for a given customer. It's often difficult or impossible for a provider to tell the difference between a router that is fully converged but running near 100% CPU after a reload from one that is thrashing and causing delays in convergence and customer traffic impacts while it runs at 100% CPU after a reload. Even worse, a scheduled or known outage on one router may trigger unplanned outages on other high-CPU devices. Even in unplanned outages, communication regarding impacts and duration is key, and these sorts of scale issues make it difficult to predict the impacts.

6.7. High Availability and Network Resiliency

In many cases, L3VPN services are carrying significant amounts of business-critical data. Customers and carriers design their networks to be robust enough to absorb single and sometimes even dual faults with little or no impact to the network as a whole. However, the expectations as to the frequency and duration of outages due to either scheduled or unscheduled events continue to go higher and higher. This is leading more providers to adopt features such as

Non-Stop Forwarding and Non-Stop Routing, as well as things like In-Service Software Upgrades to improve the chances that outages will be transparent to the underlying customers, networks, and applications using the network elements. As these become more common within the L3VPN space, they must be considered when evaluating PE scale. Often, the machinery necessary to make these reliability enhancements work requires duplication and sharing of state between multiple elements. At its most basic level, this state sharing takes more resources and more time the more state there is to be shared, so increases in the different scaling vectors discussed in this document will cause proportional increases in the complexity and resource requirements necessary for the combined feature set. In more complex scenarios and implementations, it may contribute to the complexity associated with capacity planning, and may make the response even more non-deterministic as scale increases.

6.8. New methods of horizontal scaling

When this document was being written, there was considerable discussion around the area of Software Defined Networking and Openflow[ONF]. These are technologies which provide a way to offload some of the more complex control plane elements to a more central controller device, which then programs the routing elements for correct forwarding plane operation. This is interesting in solving a problem such as described in this document because it effectively decouples the growth of the control plane from the growth of the forwarding plane. In other words, it would be possible to continue allocating more and more CPU resources to the high-overhead control plane elements discussed above, and keep it almost totally independent from the physical forwarding plane resources necessary. While in some ways this would simply move the need for horizontal scaling elsewhere, rather than actually reducing the scaling considerations, the benefit is that an SP could use commodity compute hardware, which would potentially be a lower cost and more easily scaled than your average PE router's CPU. The application of SDN/Openflow or any other interface to the routing system that offloads some control plane elements for improved BGP VPN scale is beyond the scope of this document, but may be a valid use case for future discussion within the IETF.

7. To-Do list

RFC EDITOR: Please remove this section before publication.

Still not discussed in the document:

Inter-AS VPN NNI scaling considerations (separate discussions on 10A, 10B/hybrid, 10C?) - include discussion on number of VRFs per NNI, routes per VRF, NNIs per router

Label Exhaustion

BGP Fast External Fallover

additional scaling considerations if using L2TPv3 or RSVP-TE tunneling for PE-PE transport

Future scaling considerations (MPLS-TP at the edge, interworking with L2 technologies, significant increases in density, etc)

8. Acknowledgements

The idea for this draft came from a presentation made by Ning So during the CDNI working group meeting at IETF 81 in Quebec City where some of these same scaling considerations are discussed. Thanks also to Yakov Rekhter, Luay Jalil, Jeff Loughridge, Stephane Litkowski, Rajiv Asati, and Daniel Cohn for their reviews and comments.

9. IANA Considerations

This draft makes no request to IANA..

10. Security Considerations

Security considerations for IP VPNs are covered in the protocol definitions. This draft does not introduce any new security considerations, but it is worth noting that attack vectors that result in minor impacts in a low-scale environment may make the problems observed in a high-scale or resource-constrained environment worse, thereby magnifying the potential for impacts.

11. Informative References

[EIGRP] Wikipedia.org, "Enhanced Interior Gateway Routing Protocol", <http://en.wikipedia.org/wiki/Enhanced_Interior_Gateway_Routing_Protocol>.

[I-D.ietf-idr-rfd-usable] Pelsser, C., Bush, R., Patel, K., Mohapatra, P., and O. Maennel, "Making Route Flap Damping Usable", draft-ietf-idr-rfd-usable-04 (work in progress), October 2013.

- [I-D.shishio-grow-isp-rfd-implement-survey]
Tsuchiya, S., Kawamura, S., Bush, R., and C. Pelsser,
"Route Flap Damping Deployment Status Survey", draft-
shishio-grow-isp-rfd-implement-survey-05 (work in
progress), June 2012.
- [IEEE802.1]
IEEE, "Connectivity Fault Management",
<[http://standards.ieee.org/getieee802/download/
802.1ag-2007.pdf](http://standards.ieee.org/getieee802/download/802.1ag-2007.pdf)>.
- [IEEE802.3]
IEEE, "Carrier Sense Multiple Access with Collision
Detection (CSMA/CD) Access Method and Physical Layer
Specifications",
<<http://standards.ieee.org/about/get/802/802.3.html>>.
- [ISO13239]
ISO, "High-level Data Link Control protocol",
<[http://read.pudn.com/downloads79/doc/comm/306220/
ISO%2013239.pdf](http://read.pudn.com/downloads79/doc/comm/306220/ISO%2013239.pdf)>.
- [ONF]
ONF, "The Open Networking Foundation", <[https://
www.opennetworking.org/](https://www.opennetworking.org/)>.
- [RFC1661] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51,
RFC 1661, July 1994.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway
Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private
Networks (VPNs)", RFC 4364, February 2006.
- [RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route
Reflection: An Alternative to Full Mesh Internal BGP
(IBGP)", RFC 4456, April 2006.
- [RFC4577] Rosen, E., Psenak, P., and P. Pillay-Esnault, "OSPF as the
Provider/Customer Edge Protocol for BGP/MPLS IP Virtual
Private Networks (VPNs)", RFC 4577, June 2006.
- [RFC4659] De Clercq, J., Ooms, D., Carugi, M., and F. Le Faucheur,
"BGP-MPLS IP Virtual Private Network (VPN) Extension for
IPv6 VPN", RFC 4659, September 2006.
- [RFC4781] Rekhter, Y. and R. Aggarwal, "Graceful Restart Mechanism
for BGP with MPLS", RFC 4781, January 2007.

- [RFC4984] Meyer, D., Zhang, L., and K. Fall, "Report from the IAB Workshop on Routing and Addressing", RFC 4984, September 2007.
- [RFC5838] Lindem, A., Mirtorabi, S., Roy, A., Barnes, M., and R. Aggarwal, "Support of Address Families in OSPFv3", RFC 5838, April 2010.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.
- [RFC6037] Rosen, E., Cai, Y., and IJ. Wijnands, "Cisco Systems' Solution for Multicast in BGP/MPLS IP VPNs", RFC 6037, October 2010.
- [RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.
- [RFC6565] Pillay-Esnault, P., Moyer, P., Doyle, J., Ertekin, E., and M. Lundberg, "OSPFv3 as a Provider Edge to Customer Edge (PE-CE) Routing Protocol", RFC 6565, June 2012.
- [RFC7024] Jeng, H., Uttaro, J., Jalil, L., Decraene, B., Rekhter, Y., and R. Aggarwal, "Virtual Hub-and-Spoke in BGP/MPLS VPNs", RFC 7024, October 2013.
- [Y.1731] ITU-T, "OAM functions and mechanisms for Ethernet based networks", <<http://www.itu.int/rec/T-REC-Y.1731/en>>.

Authors' Addresses

Wesley George
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Phone: +1 703-561-2540
Email: wesley.george@twcable.com

Rob Shakir
BT
London
UK

Phone: +
Email: rob.shakir@bt.com

INTERNET-DRAFT
Intended Status: Proposed Standard
Expires: 2014-04-12

Saud Asif
AT&T
Andy Green
BT
Sameer Gulrajani
Cisco
Pradeep Jain
Alcatel-Lucent
Jeffrey Zhang
Juniper
2013-10-12

MPLS/BGP Layer 3 VPN Multicast
Management Information Base

draft-ietf-l3vpn-mvpn-mib-04

Abstract

This memo defines an portion of the Management Information Base (MIB) for use with network management protocols in the Internet community.

In particular, it describes managed objects to configure and/or monitor multicast in MPLS/BGP-based Layer-3 VPN (MVPN) on an MVPN router.

Status of this Memo

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

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

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

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

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

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

Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

Table of Contents

0	Draft history	3
1	Introduction	3
1.1	Terminology	3
2	MVPN MIB	4
2.1	Summary of MIB Module	4
2.2	MIB Module Definitions	5
3	Security Considerations	30
4	IANA Considerations	30
5	Acknowledgement	30
6	References	30
6.1	Normative References	30
6.2	Informative References	31
	Authors' Addresses	31

0 Draft history

This draft is a first pass at a MIB document for [MVPN]. As such, it should be considered as a early work.

Some aspects of BGP-MVPN (see definition below in "Introduction"), such as exranet, may be specified in future revisions.

[note to author/reviewers: conformance groups to be added]

[this section should be removed as soon as its stops being relevant]

1 Introduction

Multicast in MPLS/BGP L3 VPNs is specified in {[MVPN], [BGP-MVPN]}. These specifications support either PIM or BGP as the protocol for exchanging VPN multicast (referred to as C-multicast states, where 'C-' stands for 'VPN Customer-') among PEs. In the rest of this document we'll use the term "PIM-MVPN" to refer to {[MVPN], [BGP-MVPN]} with PIM being used for exchanging C-multicast states, and "BGP-MVPN" to refer to {[MVPN], [BGP-MVPN]} with BGP is used for exchanging C-multicast states.

This document defines a standard MIB for MVPN-specific objects that are generic to both PIM-MVPN and BGP-MVPN.

This document borrowed some text from Cisco PIM-MVPN MIB [CISCO-MIB]. For PIM-MVPN this document attempts to provide coverage comparable to [CISCO-MIB], but in a generic way that applies to both PIM-MVPN and BGP-MVPN.

Comments should be made directly to the Layer-3 VPN (L3VPN) WG at l3vpn@ietf.org.

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].

This document adopts the definitions, acronyms and mechanisms described in [MVPN] and other documents that [MVPN] refers to. Familiarity with Multicast, MPLS, L3VPN, MVPN concepts and/or mechanisms is assumed.

Interchangeably, the term MVRF and MVPN are used to refer to a partiular Multicast VPN instantiation on a particular PE device.

2 MVPN MIB

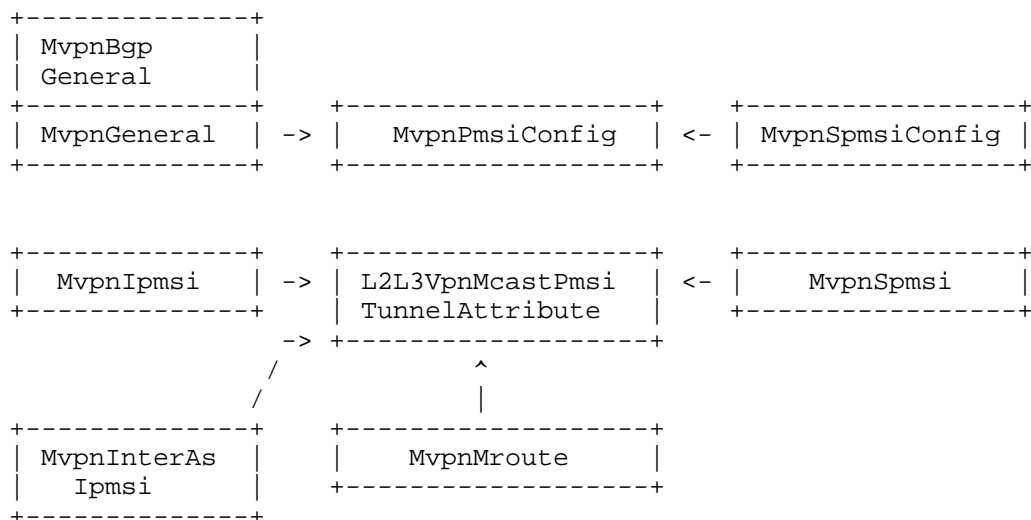
This MIB enables configuring and/or monitoring of MVPNs on PE devices: the whole multicast VPN machinery and the per-MVRFs information, including the configuration, status and operational details, such as different PMSIs and the provider tunnels implementing them.

2.1 Summary of MIB Module

The configuration and states specific to an MVPN include the following:

- C-multicast routing exchange protocol (PIM or BGP)
- I-PMSI, S-PMSI and corresponding provider tunnels
- Mapping of c-multicast states to PMSI/tunnels

To represent them, the following tables are defined.



- mvpnGeneralTable/Entry

An entry in this table is created for every MVRF in the device, for general configuration/states of the MVRF, including I-PMSI configuration.

Existence of the corresponding VRF in [L3VPN-MIB] is necessary for

a row to exist in this table.

- mvpnBgpGeneralTable/Entry

This table augments mvpnGeneralTable and is for BGP-MVPN specific information.

- mvpnSpmsiConfigTable/Entry

This table contains objects for S-PMSI configurations in an MVRF.

- mvpnPmsiConfigTable/Entry

Both I-PMSI configuration (in mvpnGeneralEntry) and S-PMSI configuration (in mvpnSpmsiConfigEntry) refer to entries in this table.

- mvpnIpmsiTable/Entry

This table contains all advertised or received intra-as I-PMSIs. With PIM-MVPN, it is applicable only when BGP-Based Autodiscovery of MVPN Membership is used.

- mvpnInterAsIpmsiTable/Entry

This table contains all advertised or received inter-as I-PMSIs. With PIM-MVPN, it is applicable only when BGP-Based Autodiscovery of MVPN Membership is used.

- mvpnSpmsiTable/Entry

This table contains all advertised or received S-PMSIs.

- l2l3VpnMcastPmsiTunnelAttributeTable/Entry

This table is defined separately in l2l3VpnMcastMIB [L2L3MVPN-MIB], which is common for both VPLS Multicast and MVPN. It contains sent/received PMSI attribute entries referred to by mvpnIpmsiEntry, mvpnSpmsiEntry, mvpnInterAsIpmsiEntry, and other MIB objects (e.g., VPLS Multicast ones).

- mvpnMrouteTable/Entry

This table augments ipMcastMIB.ipMcast.ipMcastRouteTable, for some MVPN specific information.

2.2 MIB Module Definitions

```
MCAST-VPN-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
    experimental, Unsigned32
        FROM SNMPv2-SMI

    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF

    TruthValue, RowPointer, RowStatus, TimeStamp, TimeInterval
        FROM SNMPv2-TC

    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB

    InetAddress, InetAddressType
        FROM INET-ADDRESS-MIB

    MplsLabel
        FROM MPLS-TC-STD-MIB

    mplsL3VpnVrfName, MplsL3VpnRouteDistinguisher
        FROM MPLS-L3VPN-STD-MIB

    ipMcastRouteEntry
        FROM IPMCAST-MIB

    L2L3VpnMcastProviderTunnelType
        FROM L2L3-VPN-MCAST-MIB;

mvpnMIB MODULE-IDENTITY
    LAST-UPDATED "201301071200Z" -- 07 January 2013 12:00:00 GMT
    ORGANIZATION "IETF Layer-3 Virtual Private
        Networks Working Group."
    CONTACT-INFO
        " Jeffrey (Zhaohui) Zhang
          zzhang@juniper.net

          Comments and discussion to l3vpn@ietf.org"

    DESCRIPTION
        "This MIB contains managed object definitions for
        multicast in BGP/MPLS IP VPNs defined by [MVPN].
        Copyright (C) The Internet Society (2013)."
```

-- Revision history.

```
REVISION "201301071200Z" -- 07 January 2013 12:00:00 GMT
```

```
DESCRIPTION
    "Initial version of the draft."
    ::= { experimental 99 } -- number to be assigned

-- Top level components of this MIB.
mvpnNotifications OBJECT IDENTIFIER ::= { mvpnMIB 0 }

-- tables, scalars
mvpnObjects          OBJECT IDENTIFIER ::= { mvpnMIB 1 }

-- conformance information
mvpnConformance     OBJECT IDENTIFIER ::= { mvpnMIB 2 }

-- mvpn Objects

mvpnScalars          OBJECT IDENTIFIER ::= { mvpnObjects 1 }
mvpnGeneral           OBJECT IDENTIFIER ::= { mvpnObjects 2 }
mvpnConfig            OBJECT IDENTIFIER ::= { mvpnObjects 3 }
mvpnStates            OBJECT IDENTIFIER ::= { mvpnObjects 4 }

-- Scalar Objects

mvpnMvrfNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of MVRFs for IPv4 or IPv6 or mLDP C-Multicast
         that are present in this device."
    ::= { mvpnScalars 1 }

mvpnMvrfNumberV4 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of MVRFs for IPv4 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 2 }

mvpnMvrfNumberV6 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of MVRFs for IPv6 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 3 }
```

```
mvpnMvrfNumberPimV4 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of PIM-MVPN MVRFs for IPv4 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 4 }

mvpnMvrfNumberPimV6 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of PIM-MVPN MVRFs for IPv6 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 5 }

mvpnMvrfNumberBgpV4 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of BGP-MVPN MVRFs for IPv4 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 6 }

mvpnMvrfNumberBgpV6 OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of BGP-MVPN MVRFs for IPv6 C-Multicast that are present
         in this device."
    ::= { mvpnScalars 7 }

mvpnMvrfNumberMldp OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of BGP-MVPN MVRFs for mLDP C-Multicast that are present
         in this device."
    ::= { mvpnScalars 8 }

mvpnNotificationEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-write
```



```

STATUS          current
DESCRIPTION
    "If this object is TRUE, then the generation of all
    notifications defined in this MIB is enabled."
DEFVAL { false }
::= { mvpnScalars 9 }

-- General MVRF Information Table

mvpnGeneralTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MvpnGeneralEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table specifies the general information about the MVRFs
        present in this device."
    ::= { mvpnGeneral 1 }

mvpnGeneralEntry OBJECT-TYPE
    SYNTAX          MvpnGeneralEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created for every MVRF in the
        device."
    INDEX           { mplsL3VpnVrfName }
    ::= { mvpnGeneralTable 1 }

MvpnGeneralEntry ::= SEQUENCE {
    mvpnGenOperStatusChange      INTEGER,
    mvpnGenOperChangeTime       TimeStamp,
    mvpnGenCmcastRouteProtocolV4 INTEGER,
    mvpnGenCmcastRouteProtocolV6 INTEGER,
    mvpnGenIpmsiConfigV4        RowPointer,
    mvpnGenIpmsiConfigV6        RowPointer,
    mvpnGenInterAsPmsiConfigV4  RowPointer,
    mvpnGenInterAsPmsiConfigV6  RowPointer,
    mvpnGenRowStatus             RowStatus
}

mvpnGenOperStatusChange OBJECT-TYPE
    SYNTAX          INTEGER { createdMvrf(1),
                             deletedMvrf(2),
                             modifiedMvrfIpmsiConfig(3),
                             modifiedMvrfSpmsiConfig(4)
                           }
    MAX-ACCESS      read-only
    STATUS          current

```

DESCRIPTION

"This object describes the last operational change that happened for the given MVRF.

```
createdMvrf - indicates that the MVRF was created in the
device.
```

deletedMvrf - indicates that the MVRF was deleted from the device. A row in this table will never have mvpnGenOperStatusChange equal to deletedMvrf(2), because in that case the row itself will be deleted from the table. This value for mvpnGenOperStatusChange is defined mainly for use in mvpnMvrfChange notification.

modifiedMvrfIpmsiConfig - indicates that the I-PMSI for the MVRF was configured, deleted or changed.

modifiedMvrfSpmsiConfig - indicates that the S-PMSI for the MVRF was configured, deleted or changed."

```
DEFVAL { createdMvrf }
::= { mvpnGeneralEntry 1 }
```

mvpnGenOperChangeTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The time at which the last operational change for the MVRF in question took place. The last operational change is specified by `mvprnGenOperStatusChange`."

$$::= \{ \text{mvpnGeneralEntry } 2 \}$$

mvpnGenCmcastRouteProtocolV4 OBJECT-TYPE

```
SYNTAX      INTEGER { pim (1),
                    bgp (2)
                    }
```

MAX-ACCESS read-write

```
STATUS      current
```

DESCRIPTION

"Protocol used to signal IPv4 C-multicast states across the provider core.

```
pim(1): PIM (PIM-MVPN).
```

```
bgp ( 2 ) : BGP ( BGP-MVPN ) . "
```

```
 ::= { mvpnGeneralEntry 3 }
```

mvpnGenCmcastRouteProtocolV6 OBJECT-TYPE

```
SYNTAX      INTEGER { pim (1),
                        bgp (2)
```

```
    }
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "Protocol used to signal IPv6 C-multicast states across the
        provider core.
        pim(1): PIM (PIM-MVPN).
        bgp(2): BGP (BGP-MVPN).
    ::= { mvpnGeneralEntry 4 }

mvpnGenIpmsiConfigV4 OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This points to a row in mvpnPmsiConfigTable,
        for I-PMSI configuration for IPv4."
    ::= { mvpnGeneralEntry 5 }

mvpnGenIpmsiConfigV6 OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This points to a row in mvpnPmsiConfigTable,
        for I-PMSI configuration for IPv6."
    ::= { mvpnGeneralEntry 6 }

mvpnGenInterAsPmsiConfigV4 OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This points to a row in mvpnPmsiConfigTable,
        for inter-as I-PMSI configuration for IPv4, in case of segmented
        inter-as provider tunnels."
    ::= { mvpnGeneralEntry 7 }

mvpnGenInterAsPmsiConfigV6 OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This points to a row in mvpnPmsiConfigTable,
        for inter-as I-PMSI configuration for IPv6, in case of segmented
        inter-as provider tunnels."
    ::= { mvpnGeneralEntry 8 }
```

```

mvpnGenRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION
        "This is used to create or delete a row in this table."
    ::= { mvpnGeneralEntry 9 }

-- General BGP-MVPN table

mvpnBgpGeneralTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MvpnBgpGeneralEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table augments the mvpnGeneralTable and is for BGP-MVPN
        specific information."
    ::= { mvpnGeneral 2 }

mvpnBgpGeneralEntry OBJECT-TYPE
    SYNTAX      MvpnBgpGeneralEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "The mvpnBgpGeneralEntry matches and augments an mvpnGeneralEntry
        for a BGP-MVPN instance, with BGP-MVPN specific informatoin."
    AUGMENTS    { mvpnGeneralEntry }
    ::= { mvpnBgpGeneralTable 1 }

MvpnBgpGeneralEntry ::= SEQUENCE {
    mvpnBgpGenMode          INTEGER,
    mvpnBgpGenUmhSelection  INTEGER,
    mvpnBgpGenSiteType      INTEGER,
    mvpnBgpGenCmcastImportRt MplsL3VpnRouteDistinguisher,
    mvpnBgpGenSrcAs         Unsigned32,
    mvpnBgpGenSptnlLimit    Unsigned32
}

mvpnBgpGenMode OBJECT-TYPE
    SYNTAX      INTEGER {
        rpt-spt (1),
        spt-only (2)
    }
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
        "For two different BGP-MVPN modes:
        rpt-spt(1): intersite-site shared tree mode

```

```

        spt-only(2): inter-site source-only tree mode."
 ::= { mvpnBgpGeneralEntry 1}

mvpnBgpGenUmhSelection OBJECT-TYPE
    SYNTAX          INTEGER {
                        highest-pe-address      (1),
                        c-root-group-hashing    (2),
                        ucast-umh-route         (3)
                      }
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "The UMH selection method for this mvpn, as specified in section
        5.1.3 of [MVPN]:
        highest-pe-address (1): PE with the highest address
        c-root-group-hashing (2): hashing based on (c-root, c-group)
        ucast-umh-route (3): per ucast route towards c-root"

 ::= { mvpnBgpGeneralEntry 2}

mvpnBgpGenSiteType OBJECT-TYPE
    SYNTAX          INTEGER {
                        sender-receiver (1),
                        receiver-only    (2),
                        sender-only      (3)
                      }
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "Whether this site is a receiver-only site or not.
        sender-receiver (1): both sender and receiver site.
        receiver-only (2): receiver-only site.
        sender-only (3): sender-only site."
 ::= { mvpnBgpGeneralEntry 3}

mvpnBgpGenCmcastImportRt OBJECT-TYPE
    SYNTAX          MplsL3VpnRouteDistinguisher
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "The C-multicast Import RT that this device adds to
        unicast vpn routes that it advertises for this mvpn."
 ::= { mvpnBgpGeneralEntry 4}

mvpnBgpGenSrcAs OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      read-only
    STATUS          current

```

DESCRIPTION

"The Source AS number in Source AS Extended Community that this device adds to the unicast vpn routes that it advertises for this mvpn."

```
::= { mvpnBgpGeneralEntry 5}
```

```
mvpnBgpGenSptnlLimit OBJECT-TYPE
```

```
SYNTAX          Unsigned32
```

```
MAX-ACCESS      read-write
```

```
STATUS          current
```

DESCRIPTION

"The max number of selective provider tunnels this device allows for this mvpn."

```
::= { mvpnBgpGeneralEntry 6}
```

```
-- PMSI Configuration Table
```

```
mvpnPmsiConfigTable OBJECT-TYPE
```

```
SYNTAX          SEQUENCE OF MvpnPmsiConfigEntry
```

```
MAX-ACCESS      not-accessible
```

```
STATUS          current
```

DESCRIPTION

"This table specifies the configured PMSIs."

```
::= { mvpnConfig 1 }
```

```
mvpnPmsiConfigEntry OBJECT-TYPE
```

```
SYNTAX          MvpnPmsiConfigEntry
```

```
MAX-ACCESS      not-accessible
```

```
STATUS          current
```

DESCRIPTION

"An entry in this table is created for each PMSI configured on this router. It can be referred to by either I-PMSI configuration (in mvpnGeneralEntry) or S-PMSI configuration (in mvpnSpmsiConfigEntry)"

```
INDEX          { mvpnPmsiConfigTunnelType,
                  mvpnPmsiConfigTunnelAuxInfo,
                  mvpnPmsiConfigTunnelPimGroupAddressType,
                  mvpnPmsiConfigTunnelPimGroupAddress,
                  mvpnPmsiConfigTunnelOrTemplateName }
```

```
::= { mvpnPmsiConfigTable 1 }
```

```
MvpnPmsiConfigEntry ::= SEQUENCE {
```

mvpnPmsiConfigTunnelType	L2L3VpnMcastProviderTunnelType,
mvpnPmsiConfigTunnelAuxInfo	Unsigned32,
mvpnPmsiConfigTunnelPimGroupAddressType	InetAddressType,
mvpnPmsiConfigTunnelPimGroupAddress	InetAddress,
mvpnPmsiConfigTunnelOrTemplateName	SnmpAdminString,
mvpnPmsiConfigEncapsType	INTEGER,
mvpnPmsiConfigRowStatus	RowStatus

```
}

mvpnPmsiConfigTunnelType OBJECT-TYPE
    SYNTAX          L2L3VpnMcastProviderTunnelType
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "Type of tunnel used to instantiate the PMSI."
    ::= { mvpnPmsiConfigEntry 1 }

mvpnPmsiConfigTunnelAuxInfo OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "Additional tunnel information depending on the type.
         pim:          In case of S-PMSI, number of groups starting at
                        mvpnPmsiConfigTunnelPimGroupAddress.
                        This allows a range of PIM provider tunnel
                        group addresses to be specified in S-PMSI case.
                        In I-PMSI case, it must be 1.
         rsvp-p2mp:    1 for statically specified rsvp-p2mp tunnel
                        2 for dynamically created rsvp-p2mp tunnel
         ingress-replication:
                        1 for using any existing p2p/mp2p lsp
                        2 for dynamically creating new p2p lsp"
    ::= { mvpnPmsiConfigEntry 2 }

mvpnPmsiConfigTunnelPimGroupAddressType OBJECT-TYPE
    SYNTAX          InetAddressType
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "In case of PIM provider tunnel, the type of tunnel address."
    ::= { mvpnPmsiConfigEntry 3 }

mvpnPmsiConfigTunnelPimGroupAddress OBJECT-TYPE
    SYNTAX          InetAddress
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "In case of PIM provider tunnel, the provider tunnel address."
    ::= { mvpnPmsiConfigEntry 4 }

mvpnPmsiConfigTunnelOrTemplateName OBJECT-TYPE
    SYNTAX          SnmpAdminString
    MAX-ACCESS      not-accessible
    STATUS          current
```

DESCRIPTION

"The tunnel name or template name used to create tunnels.
Depending on mvpnPmsiConfigTunnelType and
mvpnPmsiConfigTunnelAuxInfo:

dynamically created rsvp-p2mp tunnel:	template name
statically specified rsvp-p2mp tunnel:	tunnel name
ingress-replication using	
dynamically created lsps:	template name
other:	null

::= { mvpnPmsiConfigEntry 5 }

mvpnPmsiConfigEncapsType OBJECT-TYPE

SYNTAX INTEGER { greIp (1),
 ipIp (2),
 mpls (3)
 }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The encapsulation type to be used, in case of PIM tunnel or
ingress-replication."

::= { mvpnPmsiConfigEntry 6 }

mvpnPmsiConfigRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Used to create/modify/delete a row in this table."

::= { mvpnPmsiConfigEntry 7 }

-- S-PMSI configuration table

mvpnSpmsiConfigTable OBJECT-TYPE

SYNTAX SEQUENCE OF MvpnSpmsiConfigEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table specifies S-PMSI configuration."

::= { mvpnConfig 2 }

mvpnSpmsiConfigEntry OBJECT-TYPE

SYNTAX MvpnSpmsiConfigEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry is created for each S-PMSI configuration."


```

INDEX      {  mplsL3VpnVrfName,
               mvpnSpmsiConfigCmcastAddressType,
               mvpnSpmsiConfigCmcastGroupAddress,
               mvpnSpmsiConfigCmcastGroupPrefixLen,
               mvpnSpmsiConfigCmcastSourceAddress,
               mvpnSpmsiConfigCmcastSourcePrefixLen }
 ::= { mvpnSpmsiConfigTable 1 }

MvpnSpmsiConfigEntry ::= SEQUENCE {
    mvpnSpmsiConfigCmcastAddressType      InetAddressType,
    mvpnSpmsiConfigCmcastGroupAddress     InetAddress,
    mvpnSpmsiConfigCmcastGroupPrefixLen   Unsigned32,
    mvpnSpmsiConfigCmcastSourceAddress    InetAddress,
    mvpnSpmsiConfigCmcastSourcePrefixLen  Unsigned32,
    mvpnSpmsiConfigThreshold              Unsigned32,
    mvpnSpmsiConfigPmsiPointer             RowPointer,
    mvpnSpmsiConfigRowStatus              RowStatus
}

mvpnSpmsiConfigCmcastAddressType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "Type of C-multicast address"
    ::= { mvpnSpmsiConfigEntry 1 }

mvpnSpmsiConfigCmcastGroupAddress OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "C-multicast group address"
    ::= { mvpnSpmsiConfigEntry 2 }

mvpnSpmsiConfigCmcastGroupPrefixLen OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "C-multicast group address prefix length.
         A group 0 (or ::0) with prefix length 32 (or 128)
         indicates wildcard group, while a group 0 (or ::0)
         with prefix length 0 indicates any group."
    ::= { mvpnSpmsiConfigEntry 3 }

mvpnSpmsiConfigCmcastSourceAddress OBJECT-TYPE
    SYNTAX      InetAddress

```

```
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "C-multicast source address"
 ::= { mvpnSpmsiConfigEntry 4 }

mvpnSpmsiConfigCmcastSourcePrefixLen OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "C-multicast source address prefix length.
     A source 0 (or ::0) with prefix length 32 (or 128)
     indicates a wildcard source, while a source 0 (or ::0)
     with prefix length 0 indicates any source."
 ::= { mvpnSpmsiConfigEntry 5 }

mvpnSpmsiConfigThreshold OBJECT-TYPE
SYNTAX          Unsigned32 (0..4294967295)
UNITS           "kilobits per second"
MAX-ACCESS      read-write
STATUS          current
DESCRIPTION
    "The bandwidth threshold value which when exceeded for a
     multicast routing entry in the given MVRFB, triggers usage
     of S-PMSI."
 ::= { mvpnSpmsiConfigEntry 6 }

mvpnSpmsiConfigPmsiPointer OBJECT-TYPE
SYNTAX          RowPointer
MAX-ACCESS      read-write
STATUS          current
DESCRIPTION
    "This points to a row in mvpnPmsiConfigTable,
     to specify tunnel attributes."
 ::= { mvpnSpmsiConfigEntry 7 }

mvpnSpmsiConfigRowStatus OBJECT-TYPE
SYNTAX          RowStatus
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Used to create/modify/delete a row in this table."
 ::= { mvpnSpmsiConfigEntry 8 }

-- Table of intra-as I-PMSIs advertised/received

mvpnIpmsiTable OBJECT-TYPE
```

```
SYNTAX          SEQUENCE OF MvpnIpmsiEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table is for all advertised/received I-PMSI
    advertisements."
 ::= { mvpnStates 1 }

mvpnIpmsiEntry OBJECT-TYPE
SYNTAX          MvpnIpmsiEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in this table corresponds to an I-PMSI
    advertisement that is advertised/received on this router.
    This represents all the sender PEs in the MVPN,
    with the provider tunnel they use to send traffic."
INDEX { mplsL3VpnVrfName,
        mvpnIpmsiAfi,
        mvpnIpmsiRD,
        mvpnIpmsiOrigAddrType,
        mvpnIpmsiOrigAddress }
 ::= { mvpnIpmsiTable 1 }

MvpnIpmsiEntry ::= SEQUENCE {
    mvpnIpmsiAfi      Unsigned32,
    mvpnIpmsiRD       MplsL3VpnRouteDistinguisher,
    mvpnIpmsiOrigAddrType InetAddressType,
    mvpnIpmsiOrigAddress InetAddress,
    mvpnIpmsiUpTime   TimeInterval,
    mvpnIpmsiAttribute RowPointer
}

mvpnIpmsiAfi OBJECT-TYPE
SYNTAX          Unsigned32 (1|2)
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "The address family this I-PMSI is for.
    1 - IPv4
    2 - IPv6"
 ::= { mvpnIpmsiEntry 1 }

mvpnIpmsiRD OBJECT-TYPE
SYNTAX          MplsL3VpnRouteDistinguisher
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
```

```
        "The Route Distinguisher in this I-PMSI."
 ::= { mvpnIpmsiEntry 2 }

mvpnIpmsiOrigAddrType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "The Internet address type of mvpnIpmsiOrigAddress."
 ::= { mvpnIpmsiEntry 3 }

mvpnIpmsiOrigAddress OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "The BGP address of the device that originated the I-PMSI."
 ::= { mvpnIpmsiEntry 4 }

mvpnIpmsiUpTime OBJECT-TYPE
    SYNTAX      TimeInterval
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The time since this I-PMSI
         was first advertised/received by the device."
 ::= { mvpnIpmsiEntry 5 }

mvpnIpmsiAttribute OBJECT-TYPE
    SYNTAX      RowPointer
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Points to a row in the l2L3VpnMcastPmsiTunnelAttributeTable."
 ::= { mvpnIpmsiEntry 6 }

-- Table of inter-as I-PMSIs advertised/received

mvpnInterAsIpmsiTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MvpnInterAsIpmsiEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table is for all advertised/received inter-as I-PMSI
         advertisements."
 ::= { mvpnStates 2 }

mvpnInterAsIpmsiEntry OBJECT-TYPE
```

```
SYNTAX          MvpnInterAsIpmsiEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in this table corresponds to an inter-as I-PMSI
    advertisement that is advertised/received on this router.
    This represents all the ASes in the MVPN,
    with the provider tunnel used to send traffic to."
INDEX { mplsL3VpnVrfName,
        mvpnInterAsIpmsiAfi,
        mvpnInterAsIpmsiRD,
        mvpnInterAsIpmsiSrcAs }
 ::= { mvpnInterAsIpmsiTable 1 }

MvpnInterAsIpmsiEntry ::= SEQUENCE {
    mvpnInterAsIpmsiAfi      Unsigned32,
    mvpnInterAsIpmsiRD      MplsL3VpnRouteDistinguisher,
    mvpnInterAsIpmsiSrcAs    Unsigned32,
    mvpnInterAsIpmsiAttribute RowPointer
}

mvpnInterAsIpmsiAfi OBJECT-TYPE
    SYNTAX      Unsigned32 (1|2)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The address family this I-PMSI is for.
        1 - IPv4
        2 - IPv6"
    ::= { mvpnInterAsIpmsiEntry 1 }

mvpnInterAsIpmsiRD OBJECT-TYPE
    SYNTAX      MplsL3VpnRouteDistinguisher
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The Route Distinguisher in this inter-as I-PMSI."
    ::= { mvpnInterAsIpmsiEntry 2 }

mvpnInterAsIpmsiSrcAs OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The source-as in this inter-as I-PMSI."
    ::= { mvpnInterAsIpmsiEntry 3 }

mvpnInterAsIpmsiAttribute OBJECT-TYPE
```

```

SYNTAX          RowPointer
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Points to a row in the l2L3VpnMcastPmsiTunnelAttributeTable."
 ::= { mvpnInterAsIpmsiEntry 4 }

-- Table of S-PMSIs advertised/received

mvpnSpmsiTable OBJECT-TYPE
SYNTAX          SEQUENCE OF MvpnSpmsiEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table has information about the S-PMSIs sent/received
     by a device."
 ::= { mvpnStates 3 }

mvpnSpmsiEntry OBJECT-TYPE
SYNTAX          MvpnSpmsiEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in this table is created or updated for every S-PMSI
     advertised/received in a particular MVRF."
INDEX { mplsL3VpnVrfName,
        mvpnSpmsiCmcastAddrType,
        mvpnSpmsiCmcastGroup,
        mvpnSpmsiCmcastGroupPrefixLen,
        mvpnSpmsiCmcastSource,
        mvpnSpmsiCmcastSourcePrefixLen,
        mvpnSpmsiOrigAddrType,
        mvpnSpmsiOrigAddress}
 ::= { mvpnSpmsiTable 1 }

MvpnSpmsiEntry ::= SEQUENCE {
    mvpnSpmsiCmcastAddrType      InetAddressType,
    mvpnSpmsiCmcastGroup         InetAddress,
    mvpnSpmsiCmcastGroupPrefixLen Unsigned32,
    mvpnSpmsiCmcastSource        InetAddress,
    mvpnSpmsiCmcastSourcePrefixLen Unsigned32,
    mvpnSpmsiOrigAddrType        InetAddressType,
    mvpnSpmsiOrigAddress          InetAddress,
    mvpnSpmsiTunnelAttribute     RowPointer,
    mvpnSpmsiUpTime              TimeInterval,
    mvpnSpmsiExpTime              TimeInterval,
    mvpnSpmsiRefCnt              Unsigned32
}

```

mvpnSpmsiCmcastAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The Internet address type of mvpnSpmsiCmcastGroup/Source."
 ::= { mvpnSpmsiEntry 1 }

mvpnSpmsiCmcastGroup OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "S-PMSI C-multicast group address.
 If it is 0 (or ::0), this is a wildcard group,
 and mvpnSpmsiCmcastGroupPrefixLen must be 32 (or 128)."
 ::= { mvpnSpmsiEntry 2 }

mvpnSpmsiCmcastGroupPrefixLen OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "S-PMSI C-multicast group address prefix length."
 ::= { mvpnSpmsiEntry 3 }

mvpnSpmsiCmcastSource OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "S-PMSI C-multicast source address
 If it is 0 (or ::0), this is a wildcard source,
 and mvpnSpmsiCmcastSourcePrefixLen must be 32 (or 128)."
 ::= { mvpnSpmsiEntry 4 }

mvpnSpmsiCmcastSourcePrefixLen OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "S-PMSI C-multicast source address prefix length."
 ::= { mvpnSpmsiEntry 5 }

mvpnSpmsiOrigAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"The Internet address type of mvpnSpmsiOrigAddress."

::= { mvpnSpmsiEntry 6 }

mvpnSpmsiOrigAddress OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The BGP address of the device that originated the S-PMSI."

::= { mvpnSpmsiEntry 7 }

mvpnSpmsiTunnelAttribute OBJECT-TYPE

SYNTAX RowPointer

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A row pointer to the l2L3VpnMcastPmsiTunnelAttributeTable"

::= { mvpnSpmsiEntry 8 }

mvpnSpmsiUpTime OBJECT-TYPE

SYNTAX TimeInterval

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The time since this S-PMSI
was first advertised/received by the device."

::= { mvpnSpmsiEntry 9 }

mvpnSpmsiExpTime OBJECT-TYPE

SYNTAX TimeInterval

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"For UDP-based S-PMSI signaling for PIM-MVPN,
the amount of time remaining before this
received S-PMSI Join Message expires,
or the next S-PMSI Join Message refresh is to be
advertised again from the device.
Otherwise, it is 0."

::= { mvpnSpmsiEntry 10 }

mvpnSpmsiRefCnt OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of c-multicast routes that are mapped to


```
        this S-PMSI."
 ::= { mvpnSpmsiEntry 11 }

-- Table of multicast routes in an MVPN

mvpnMrouteTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MvpnMrouteEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table augments ipMcastRouteTable, to provide some MVPN
        specific information."
    ::= { mvpnStates 4 }

mvpnMrouteEntry OBJECT-TYPE
    SYNTAX          MvpnMrouteEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The mvpnMrouteEntry matches and augments an ipMcastRouteEntry,
        with MVPN specific information, such as PMSI used."
    AUGMENTS        { ipMcastRouteEntry }
    ::= { mvpnMrouteTable 1 }

MvpnMrouteEntry ::= SEQUENCE {
    mvpnMroutePmsiPointer          RowPointer,
    mvpnMrouteNumberOfLocalReplication  Unsigned32,
    mvpnMrouteNumberOfRemoteReplication Unsigned32,
    mvpnMrouteDataRate             Unsigned32
}

mvpnMroutePmsiPointer OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The I-PMSI or S-PMSI this C-multicast route is using.
        This is important because an implementation may not have an
        interface corresponding to a provider tunnel,
        that can be used in ipMcastRouteNextHopEntry."
    ::= { mvpnMrouteEntry 1 }

mvpnMrouteNumberOfLocalReplication OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of replications to local receivers."
```

```
 ::= { mvpnMrouteEntry 2 }

mvpnMrouteNumberOfRemoteReplication OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "Number of (local) replications to remote receivers."
    ::= { mvpnMrouteEntry 3 }

mvpnMrouteDataRate OBJECT-TYPE
    SYNTAX      Unsigned32 (0..4294967295)
    UNITS       "kilobits per second"
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The data rate for traffic following this route."
    ::= { mvpnMrouteEntry 4 }

-- MVPN Notifications

mvpnMvrfChange NOTIFICATION-TYPE
    OBJECTS      {
        mvpnGenOperStatusChange
    }
    STATUS      current
    DESCRIPTION
        "A mvpnMvrfChange notification signifies a change about
        a MVRF in the device. The change event can be creation of
        the MVRF, deletion of the MVRF or an update on the I-PMSI
        or S-PMSI configuration of the MVRF. The change event
        is indicated by mvpnGenOperStatusChange embedded in
        the notification. The user can then query
        mvpnGeneralTable, and/or mvpnSpmsiConfigTable to
        get the details of the change as necessary."

        Note: Since the creation of a MVRF is often followed by
        configuration of I-PMSI and/or S-PMSIs for the MVRF,
        more than one (three at most) notifications for a MVRF may
        be generated serially, and it is really not necessary to
        generate all three of them. An agent may choose to generate a
        notification for the last event only, that is for S-PMSI
        configuration.

        Similarly, deletion of I-PMSI and S-PMSI configuration on a
        MVRF happens before a MVRF is deleted and it is recommended
        that the agent send the notification for MVRF deletion
        event only."
```

```
::= { mvpnNotifications 2 }

-- MVPN MIB Conformance Information

mvpnGroups      OBJECT IDENTIFIER ::= { mvpnConformance 1 }
mvpnCompliances OBJECT IDENTIFIER ::= { mvpnConformance 2 }

-- Compliance Statements

mvpnCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "The compliance statement "
    MODULE -- this module
    MANDATORY-GROUPS {
        mvpnScalarGroup,
        mvpnGeneralGroup,
        mvpnSpmsiConfigGroup,
        mvpnSpmsiGroup,
        mvpnMrouteGroup
    }

    GROUP mvpnIpmsiGroup
    DESCRIPTION
        "This group is mandatory for systems that support
        BGP signaling for I-PMSI."

    GROUP mvpnInterAsIpmsiGroup
    DESCRIPTION
        "This group is mandatory for systems that support
        Inter-AS Segmented I-PMSI."

    GROUP mvpnBgpGeneralGroup
    DESCRIPTION
        "This group is mandatory for systems that support
        BGP-MVPN."

::= { mvpnCompliances 1 }

-- units of conformance

mvpnScalarGroup OBJECT-GROUP
    OBJECTS {
        mvpnMvrfNumber,
        mvpnMvrfNumberV4,
        mvpnMvrfNumberV6,
        mvpnMvrfNumberPimV4,
```

```
        mvpnMvrfNumberPimV6,
        mvpnMvrfNumberBgpV4,
        mvpnMvrfNumberBgpV6,
        mvpnMvrfNumberMldp,
        mvpnNotificationEnable
    }
STATUS          current
DESCRIPTION
    "These objects are used to monitor/manage
    global MVPN parameters."
::= { mvpnGroups 1 }

mvpnGeneralGroup    OBJECT-GROUP
OBJECTS {
    mvpnGenOperStatusChange,
    mvpnGenOperChangeTime,
    mvpnGenCmcastRouteProtocolV4,
    mvpnGenCmcastRouteProtocolV6,
    mvpnGenIpmsiConfigV4,
    mvpnGenIpmsiConfigV6,
    mvpnGenInterAsPmsiConfigV4,
    mvpnGenInterAsPmsiConfigV6,
    mvpnGenRowStatus
}
STATUS          current
DESCRIPTION
    "These objects are used to monitor/manage
    per-VRF MVPN parameters."
::= { mvpnGroups 2 }

mvpnPmsiConfigGroup    OBJECT-GROUP
OBJECTS {
    mvpnPmsiConfigEncapsType,
    mvpnPmsiConfigRowStatus
}
STATUS          current
DESCRIPTION
    "These objects are used to monitor/manage
    PMSI tunnel configurations."
::= { mvpnGroups 3 }

mvpnSpmsiConfigGroup    OBJECT-GROUP
OBJECTS {
    mvpnSpmsiConfigThreshold,
    mvpnSpmsiConfigPmsiPointer,
    mvpnSpmsiConfigRowStatus
}
STATUS          current
```

```
DESCRIPTION
    "These objects are used to monitor/manage
    S-PMSI configurations."
 ::= { mvpnGroups 4 }

mvpnIpmsiGroup      OBJECT-GROUP
OBJECTS {
    mvpnIpmsiUpTime,
    mvpnIpmsiAttribute
}
STATUS              current
DESCRIPTION
    "These objects are used to monitor/manage
    Intra-AS I-PMSI attributes."
 ::= { mvpnGroups 5 }

mvpnInterAsIpmsiGroup  OBJECT-GROUP
OBJECTS {
    mvpnInterAsIpmsiAttribute
}
STATUS              current
DESCRIPTION
    "These objects are used to monitor/manage
    Inter-AS I-PMSI attributes."
 ::= { mvpnGroups 6 }

mvpnSpmsiGroup      OBJECT-GROUP
OBJECTS {
    mvpnSpmsiTunnelAttribute,
    mvpnSpmsiUpTime,
    mvpnSpmsiExpTime,
    mvpnSpmsiRefCnt
}
STATUS              current
DESCRIPTION
    "These objects are used to monitor/manage
    S-PMSI attributes."
 ::= { mvpnGroups 7 }

mvpnMrouteGroup      OBJECT-GROUP
OBJECTS {
    mvpnMrouteNumberOfLocalReplication,
    mvpnMrouteNumberOfRemoteReplication,
    mvpnMrouteDataRate
}
STATUS              current
DESCRIPTION
    "These objects are used to monitor/manage
```

```
        VPN multicast forwarding states."
 ::= { mvpnGroups 8 }

mvpnBgpGeneralGroup OBJECT-GROUP
OBJECTS {
    mvpnBgpGenMode,
    mvpnBgpGenUmhSelection,
    mvpnBgpGenSiteType,
    mvpnBgpGenCmcastImportRt,
    mvpnBgpGenSrcAs,
    mvpnBgpGenSptnlLimit
}
STATUS current
DESCRIPTION
    "These objects are used to monitor/manage BGP-MVPN "
 ::= { mvpnGroups 9 }

mvpnOptionalGroup OBJECT-GROUP
OBJECTS {
    mvpnMroutePmsiPointer
}
STATUS current
DESCRIPTION
    "Support of these object is not required."
 ::= { mvpnGroups 10}
```

END

3 Security Considerations

<Security considerations text>

4 IANA Considerations

<IANA considerations text>

5 Acknowledgement

Some of the text has been taken almost verbatim from [CISCO-MIB].

We would like to thank Yakov Rekhter, Jeffrey Haas, Huajin Jeng, Durga Prasad Velamuri for their helpful comments.

6 References

6.1 Normative References

- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4382] Nadeau, T., Ed., and H. van der Linde, Ed., "MPLS/BGP Layer 3 Virtual Private Network (VPN) Management Information Base", RFC 4382, February 2006.
- [MROUTE-MIB]McWalter, D., Thaler, D., and A. Kessler, "IP Multicast MIB", RFC 5132, December 2007.
- [MVPN] Eric C. Rosen, Rahul Aggarwal, et. al., Multicast in MPLS/BGP IP VPNs, RFC 6513.
- [BGP-MVPN] R. Aggarwal, E. Rosen, T. Morin, Y. Rekhter, BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs, RFC 6514.
- [L2L3MVPN-MIB] Zhang, J., L2L3 VPN Multicast MIB, draft-zzhang-l2l3-vpn-mcast-mib, Work In Progress.

6.2 Informative References

- [CISCO-MIB] Susheela Vaidya, Thomas D. Nadeau, Harmen Van der Linde, Multicast in BGP/MPLS IP VPNs Management Information Base, draft-svaidya-mcast-vpn-mib-02.txt, Work In Progress, April 2005.

Authors' Addresses

Saud Asif
AT&T
C5-3D30
200 South Laurel Avenue
Middletown, NJ 07748
USA
Email: sasif@att.com

Andy Green
BT Design 21CN Converged Core IP & Data
01473 629360
Adastral Park, Martlesham Heath, Ipswich IP5 3RE
UK
Email: andy.da.green@bt.com

Sameer Gulrajani
Cisco Systems
Tasman Drive
San Jose, CA 95134

USA

Email: sameerg@cisco.com

Pradeep G. Jain
Alcatel-Lucent Inc
701 E Middlefield road
Mountain view, CA 94043
USA
Email: pradeep.jain@alcatel-lucent.com

Jeffrey (Zhaohui) Zhang
Juniper Networks, Inc.
10 Technology Park Drive
Westford, MA 01886
USA
Email: zzhang@juniper.net

Network working group
Internet Draft
Category: Informational

X. Xu
Huawei Technologies

S. Hares

Y. Fan
China Telecom

C. Jacquenet
France Telecom

Expires: January 2014

July 15, 2013

Virtual Subnet: A L3VPN-based Subnet Extension Solution

draft-xu-virtual-subnet-11

Abstract

This document describes a Layer3 Virtual Private Network (L3VPN)-based subnet extension solution referred to as Virtual Subnet, which can be used as a kind of Layer3 network virtualization overlay approach for data center interconnect.

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/lid-abstracts.txt>.

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

This Internet-Draft will expire on January 15, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

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].

Table of Contents

1. Introduction	4
2. Terminology	6
3. Solution Description.....	6
3.1. Unicast	6
3.1.1. Intra-subnet Unicast	6
3.1.2. Inter-subnet Unicast	7
3.2. Multicast	9
3.3. CE Host Discovery	9
3.4. ARP/ND Proxy	10
3.5. CE Host Mobility	10
3.6. Forwarding Table Scalability	10
3.6.1. MAC Table Reduction on Data Center Switches	10
3.6.2. PE Router FIB Reduction	11
3.6.3. PE Router RIB Reduction	12
3.7. ARP/ND Cache Table Scalability on Default Gateways	14
3.8. ARP/ND and Unknown Uncast Flood Avoidance	14
3.9. Path Optimization	14
4. Considerations for Non-IP traffic	15
5. Security Considerations	15
6. IANA Considerations	15
7. Acknowledgements	15
8. References	15

8.1. Normative References	15
8.2. Informative References	15
Authors' Addresses	16

1. Introduction

For business continuity purposes, Virtual Machine (VM) migration across data centers is commonly used in those situations such as data center maintenance, data center migration, data center consolidation, data center expansion, and data center disaster avoidance. It's generally admitted that IP renumbering of servers (i.e., VMs) after the migration is usually complex and costly at the risk of extending the business downtime during the process of migration. To allow the migration of a VM from one data center to another without IP renumbering, the subnet on which the VM resides needs to be extended across these data centers.

In Infrastructure-as-a-Service (IaaS) cloud data center environments, to achieve subnet extension across multiple data centers in a scalable way, the following requirements SHOULD be considered for any data center interconnect solution:

1) VPN Instance Space Scalability

In a modern cloud data center environment, thousands or even tens of thousands of tenants could be hosted over a shared network infrastructure. For security and performance isolation purposes, these tenants need to be isolated from one another. Hence, the data center interconnect solution SHOULD be capable of providing a large enough Virtual Private Network (VPN) instance space for tenant isolation.

2) Forwarding Table Scalability

With the development of server virtualization technologies, a single cloud data center containing millions of VMs is not uncommon. This number already implies a big challenge for data center switches, especially for core/aggregation switches, from the perspective of forwarding table scalability. Provided that multiple data centers of such scale were interconnected at layer2, this challenge would be even worse. Hence an ideal data center interconnect solution SHOULD prevent the forwarding table size of data center switches from growing by folds as the number of data centers to be interconnected increases. Furthermore, if any kind of L2VPN or L3VPN technologies is used for interconnecting data centers, the scale of forwarding tables on PE routers SHOULD be taken into consideration as well.

3) ARP/ND Cache Table Scalability on Default Gateways

[RFC6820] notes that the Address Resolution Protocol (ARP)/Neighbor Discovery (ND) cache tables maintained by data center default gateways in cloud data centers can raise both scalability and security issues. Therefore, an ideal data center interconnect solution SHOULD prevent the ARP/ND cache table size from growing by multiples as the number of data centers to be connected increases.

4) ARP/ND and Unknown Unicast Flood Suppression or Avoidance

It's well-known that the flooding of Address Resolution Protocol (ARP)/Neighbor Discovery (ND) broadcast/multicast and unknown unicast traffic within a large Layer2 network are likely to affect performances of networks and hosts. As multiple data centers each containing millions of VMs are interconnected together across the Wide Area Network (WAN) at layer2, the impact of flooding as mentioned above will become even worse. As such, it becomes increasingly desirable for data center operators to suppress or even avoid the flooding of ARP/ND broadcast/multicast and unknown unicast traffic across data centers.

5) Path Optimization

A subnet usually indicates a location in the network. However, when a subnet has been extended across multiple geographically dispersed data center locations, the location semantics of such subnet is not retained any longer. As a result, the traffic from a cloud user (i.e., a VPN user) which is destined for a given server located at one data center location of such extended subnet may arrive at another data center location firstly according to the subnet route, and then be forwarded to the location where the service is actually located. This suboptimal routing would obviously result in the unnecessary consumption of the bandwidth resources which are intended for data center interconnection. Furthermore, in the case where the traditional VPLS technology [RFC4761, RFC4762] is used for data center interconnect and default gateways of different data center locations are configured within the same virtual router redundancy group, the returning traffic from that server to the cloud user may be forwarded at layer2 to a default gateway located at one of the remote data center premises, rather than the one placed at the local data center location. This suboptimal routing would also unnecessarily consume the bandwidth resources which are intended for data center interconnect.

This document describes a L3VPN-based subnet extension solution referred to as Virtual Subnet (VS), which can meet all of the

requirements of cloud data center interconnect as described above. Since VS mainly reuses existing technologies including BGP/MPLS IP VPN [RFC4364] and ARP/ND proxy [RFC925][RFC1027][RFC4389], it allows those service providers offering IaaS public cloud services to interconnect their geographically dispersed data centers in a much scalable way, and more importantly, data center interconnection design can rely upon their existing MPLS/BGP IP VPN infrastructures and their experiences in the delivery and the operation of MPLS/BGP IP VPN services.

Although Virtual Subnet is described as a data center interconnection solution in this document, there is no reason to assume that this technology couldn't be used within data centers.

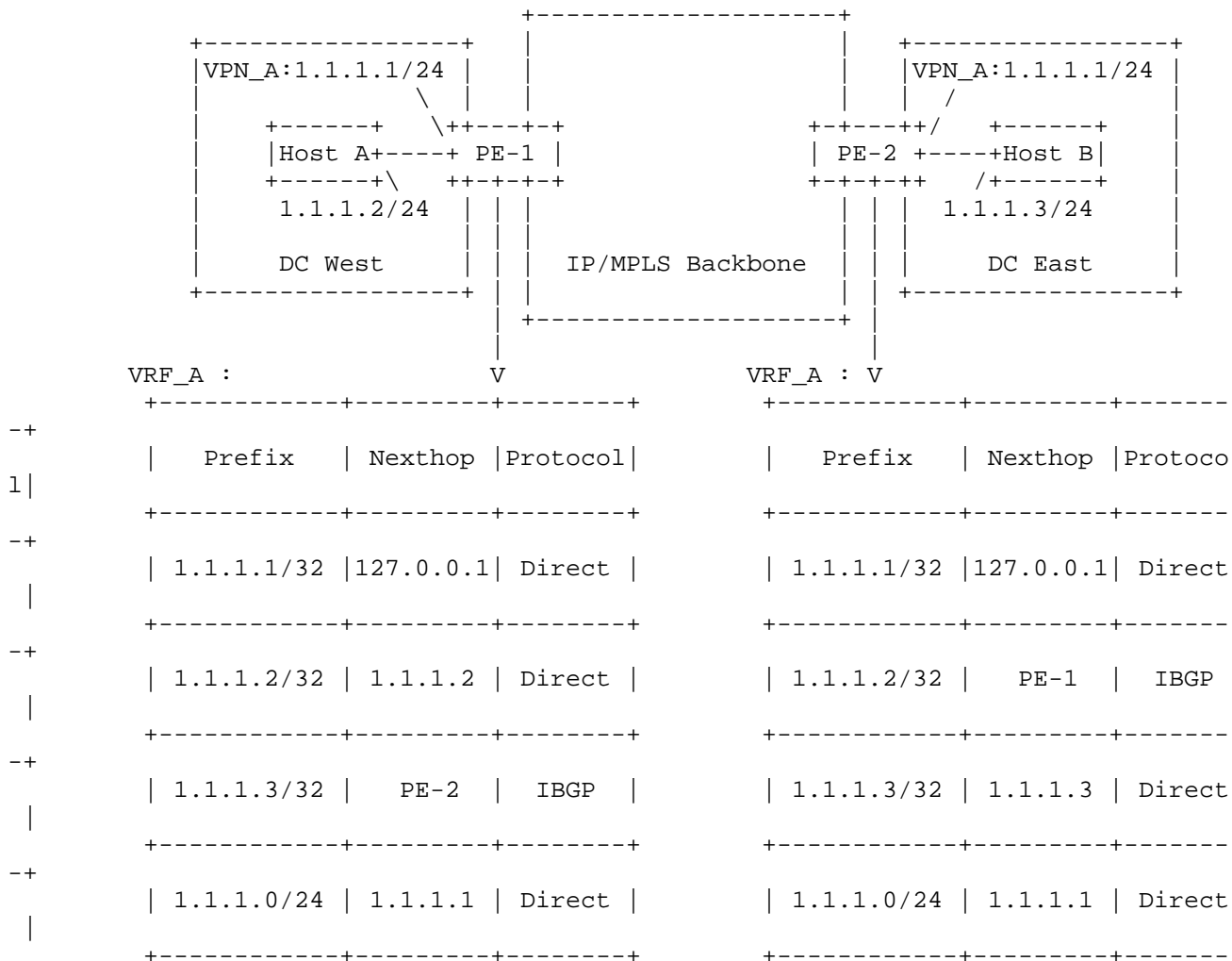
2. Terminology

This memo makes use of the terms defined in [RFC4364], [RFC2338] [MVPN] and [VA-AUTO].

3. Solution Description

3.1. Unicast

3.1.1. Intra-subnet Unicast



-+

Figure 1: Intra-subnet Unicast Example

Xu, et al.

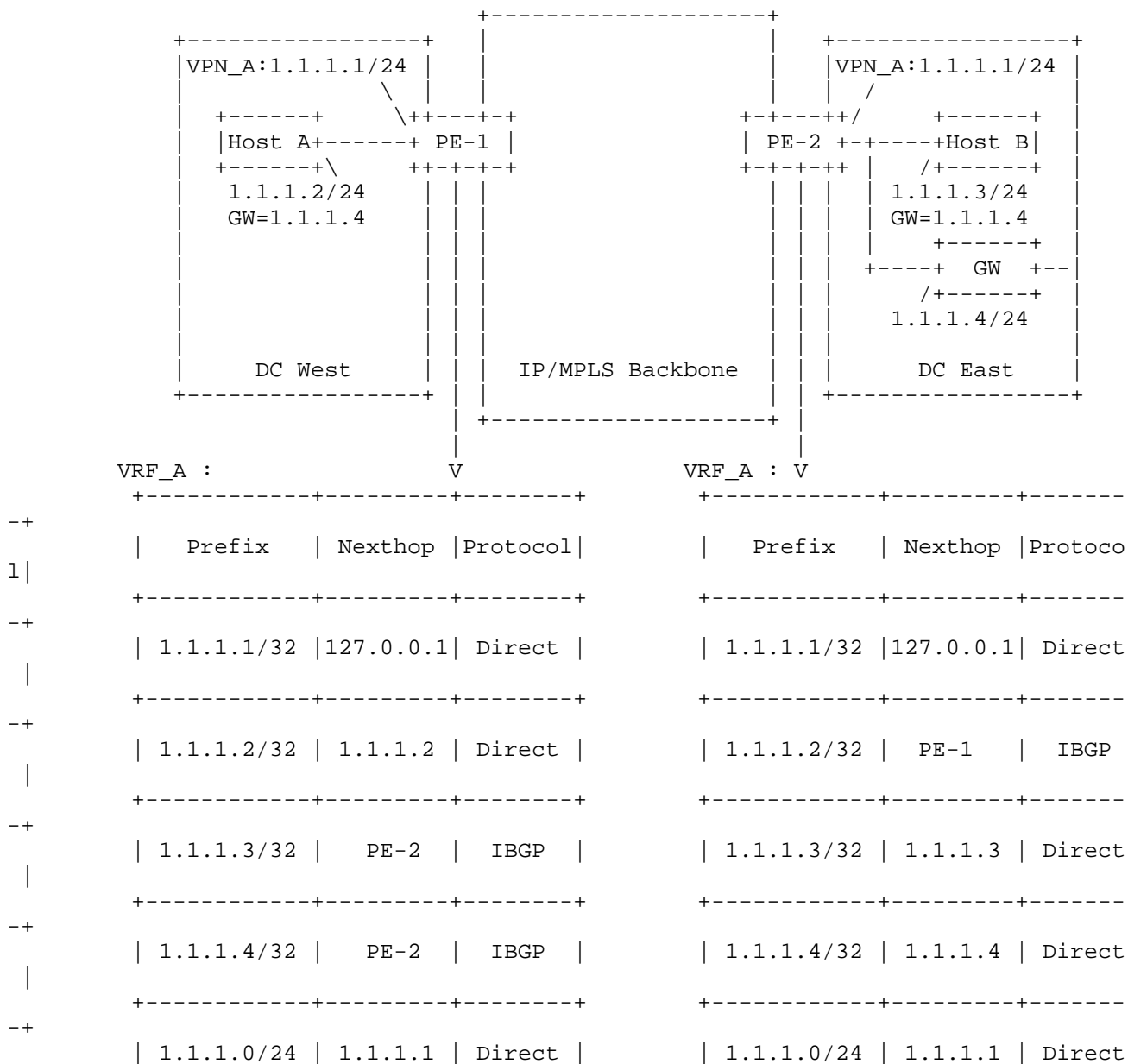
Expires January 15, 2014

[Page 6]

As shown in Figure 1, two CE hosts (i.e., Hosts A and B) belonging to the same subnet (i.e., 1.1.1.0/24) are located at different data centers (i.e., DC West and DC East) respectively. PE routers (i.e., PE-1 and PE-2) which are used for interconnecting these two data centers create host routes for their local CE hosts respectively and then advertise them via L3VPN signaling. Meanwhile, ARP proxy is enabled on VRF attachment circuits of these PE routers.

Now assume host A sends an ARP request for host B before communicating with host B. Upon receiving the ARP request, PE-1 acting as an ARP proxy returns its own MAC address as a response. Host A then sends IP packets for host B to PE-1. PE-1 tunnels such packets towards PE-2 which in turn forwards them to host B. Thus, hosts A and B can communicate with each other as if they were located within the same subnet.

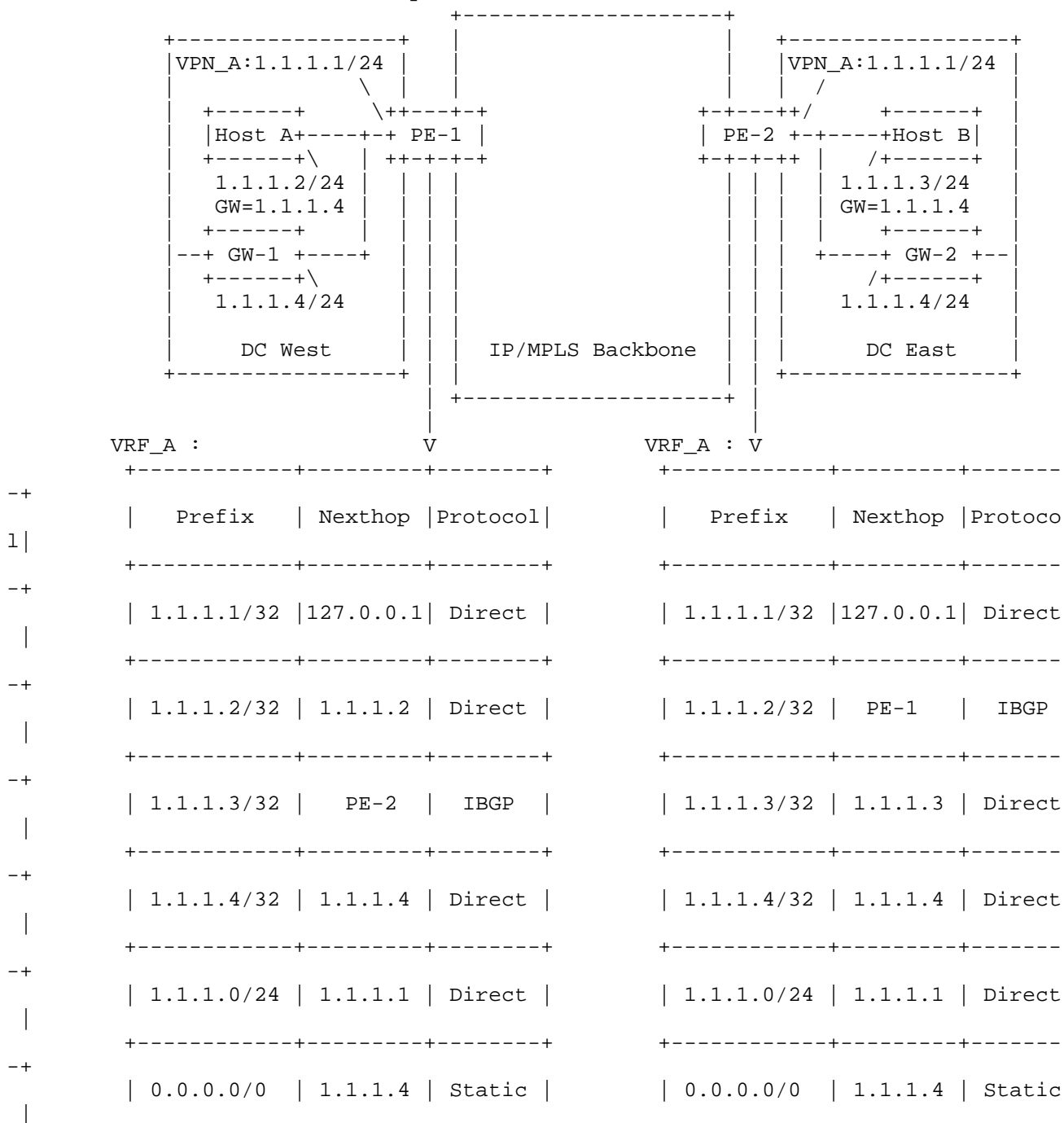
3.1.2. Inter-subnet Unicast



	+-----+-----+-----+				+-----+-----+-----			
-+	0.0.0.0/0	PE-2	IBGP		0.0.0.0/0	1.1.1.4	Static	
	+-----+-----+-----+				+-----+-----+-----			
-+								

Figure 2: Inter-subnet Unicast Example (1)

As shown in Figure 2, only one data center (i.e., DC East) is deployed with a default gateway (i.e., GW). PE-2 which is connected to GW would either be configured with or learn from GW a default route with next-hop being pointed to GW. Meanwhile, this route is distributed to other PE routers (i.e., PE-1) as per normal [RFC4364] operation. Assume host A sends an ARP request for its default gateway (i.e., 1.1.1.4) prior to communicating with a destination host outside of its subnet. Upon receiving this ARP request, PE-1 acting as an ARP proxy returns its own MAC address as a response. Host A then sends a packet for Host B to PE-1. PE-1 tunnels such packet towards PE-2 according to the default route learnt from PE-2, which in turn forwards that packet to GW.



+-----+-----+-----+ +-----+-----+-----+
-+

Figure 3: Inter-subnet Unicast Example (2)

As shown in Figure 3, in the case where each data center is deployed with a default gateway, CE hosts will get ARP responses directly from their local default gateways, rather than from their local PE routers when sending ARP requests for their default gateways.

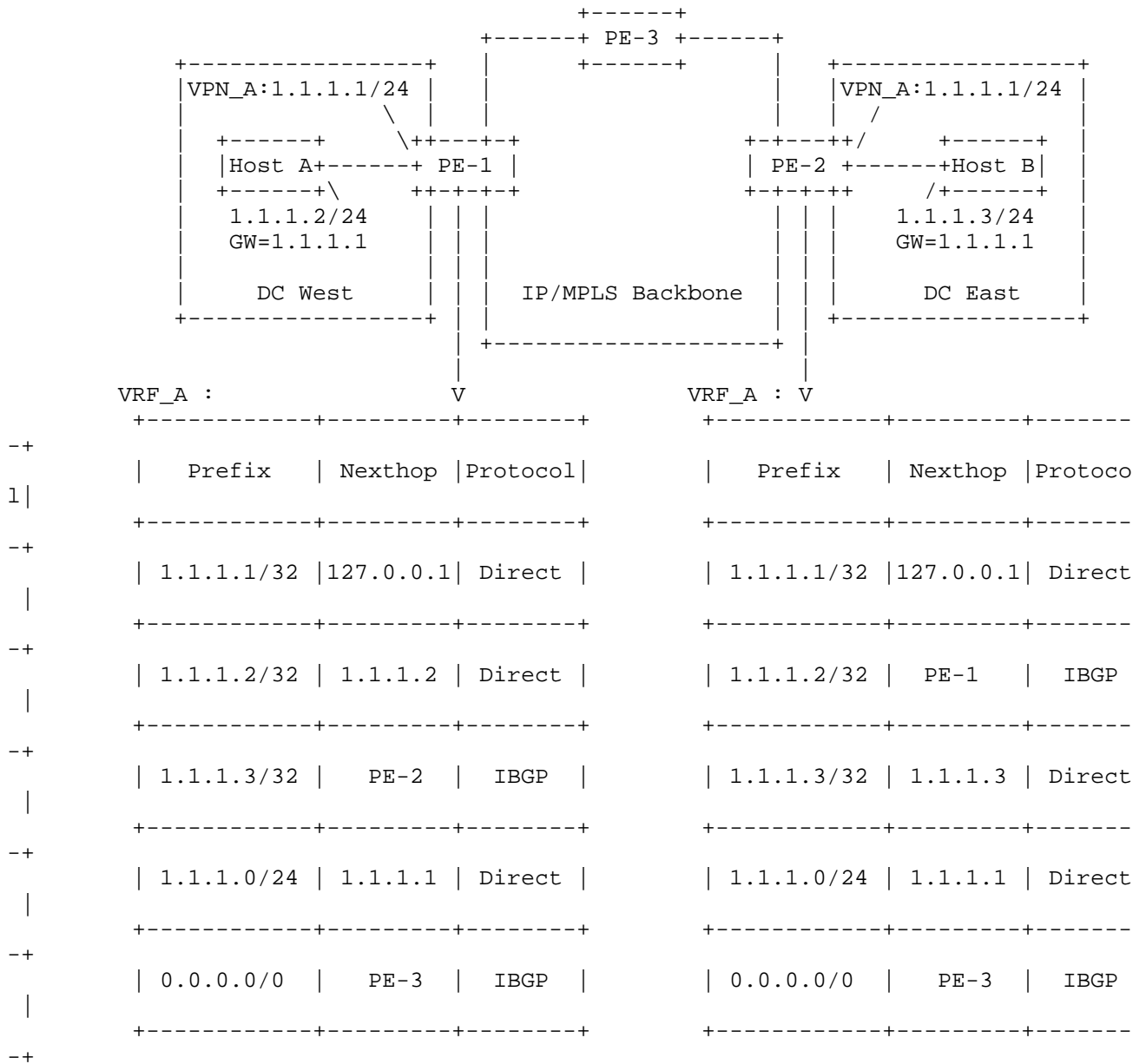


Figure 4: Inter-subnet Unicast Example (3)

Alternatively, as shown in Figure 4, PE routers themselves could be directly configured as default gateways of their locally connected CE hosts as long as these PE routers have routes for outside networks.

3.2. Multicast

To support IP multicast between CE hosts of the same virtual subnet, MVPN technology [MVPN] could be directly reused. For example, PE routers attached to a given VPN join a default provider multicast distribution tree which is dedicated for that VPN. Ingress PE routers, upon receiving multicast packets from their local CE hosts, forward them towards remote PE routers through the corresponding default provider multicast distribution tree.

More details about how to support multicast and broadcast in VS will be explored in a later version of this document.

3.3. CE Host Discovery

PE routers SHOULD be able to discover their local CE hosts and keep the list of these hosts up to date in a timely manner so as to ensure

the availability and accuracy of the corresponding host routes originated from them. PE routers could accomplish local CE host discovery by some traditional host discovery mechanisms using ARP or ND protocols. Furthermore, Link Layer Discovery Protocol (LLDP) described in [802.1AB] or VSI Discovery and Configuration Protocol (VDP) described in [802.1Qbg], or even interaction with the data center orchestration system could also be considered as a means to dynamically discover local CE hosts.

3.4. ARP/ND Proxy

Acting as ARP or ND proxies, PE routers SHOULD only respond to an ARP request or Neighbor Solicitation (NS) message for the target host when there is a corresponding host route in the associated VRF and the outgoing interface of that route is different from the one over which the ARP request or the NS message arrived.

In the scenario where a given VPN site (i.e., a data center) is multi-homed to more than one PE router via an Ethernet switch or an Ethernet network, Virtual Router Redundancy Protocol (VRRP) [RFC5798] is usually enabled on these PE routers. In this case, only the PE router being elected as the VRRP Master is allowed to perform the ARP/ND proxy function.

3.5. CE Host Mobility

During the VM migration process, the PE router to which the moving VM is now attached would create a host route for that CE host upon receiving a notification message of VM attachment while the PE router to which the moving VM was previously attached would withdraw the corresponding host route when receiving a notification message of VM detachment. Meanwhile, the latter PE router could optionally broadcast a gratuitous ARP/ND message on behalf of that CE host with source MAC address being one of its own. In the way, the ARP/ND entry of that moved CE host which has been cached on any local CE host would be updated accordingly.

3.6. Forwarding Table Scalability

3.6.1. MAC Table Reduction on Data Center Switches

In a VS environment, the MAC learning domain associated with a given virtual subnet which has been extended across multiple data centers is partitioned into segments and each segment is confined within a single data center. Therefore data center switches only need to learn local MAC addresses, rather than learning both local and remote MAC addresses.

3.6.2. PE Router FIB Reduction

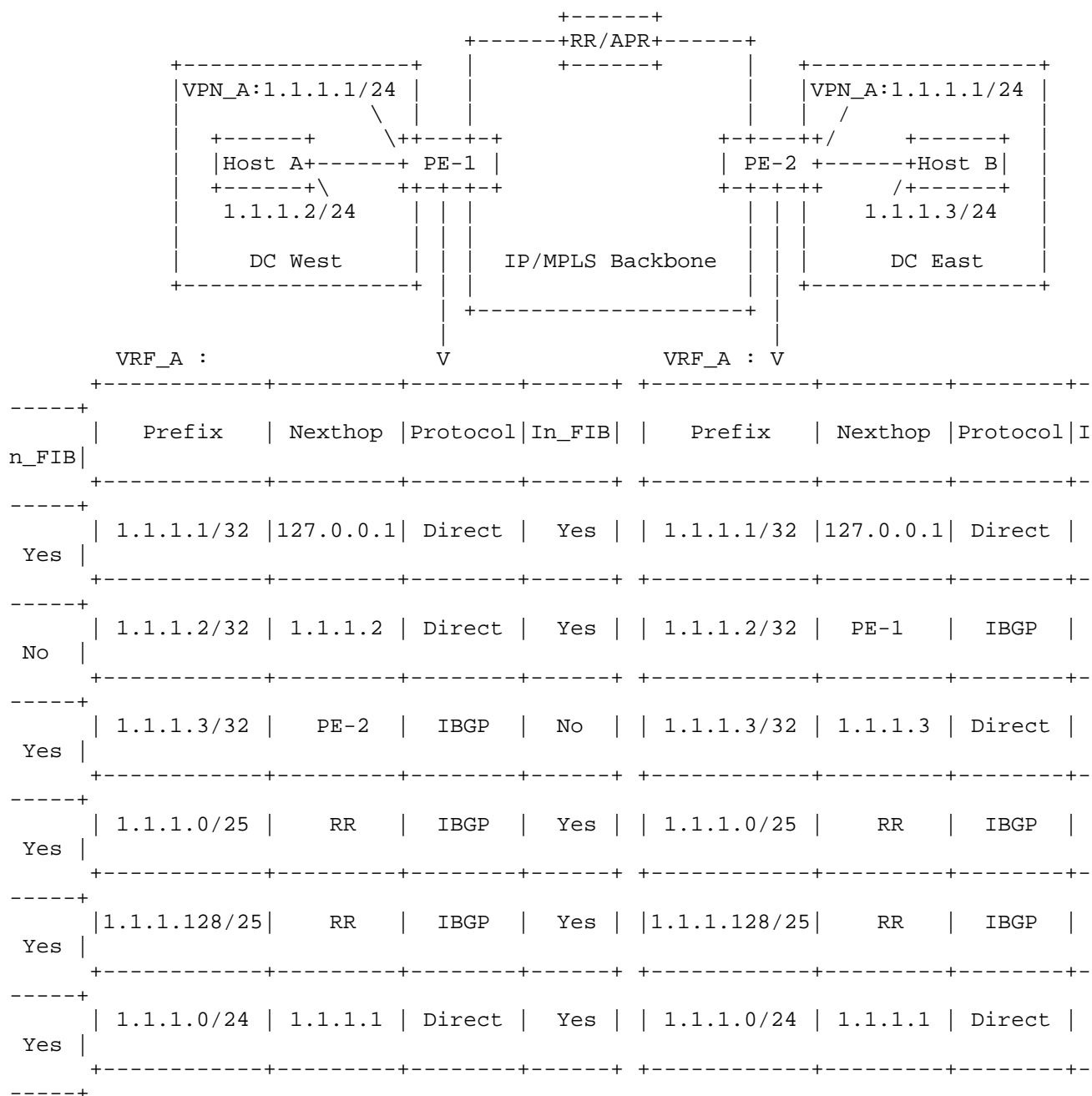


Figure 5: FIB Reduction Example

To reduce the FIB size of PE routers, Virtual Aggregation (VA) [VA-AUTO] technology can be used. Take the VPN instance A shown in Figure 5 as an example, the procedures of FIB reduction are as follows:

- 1) Multiple more specific prefixes (e.g., 1.1.1.0/25 and 1.1.1.128/25) corresponding to the prefix of virtual subnet (i.e., 1.1.1.0/24) are configured as Virtual Prefixes (VPs) and a Route-Reflector (RR) is configured as an Aggregation Point Router (APR) for these VPs. PE routers as RR clients advertise host routes for their own local CE hosts to the RR which in turn, as an APR, installs those host routes into its FIB and then attach the "can-suppress" tag to those

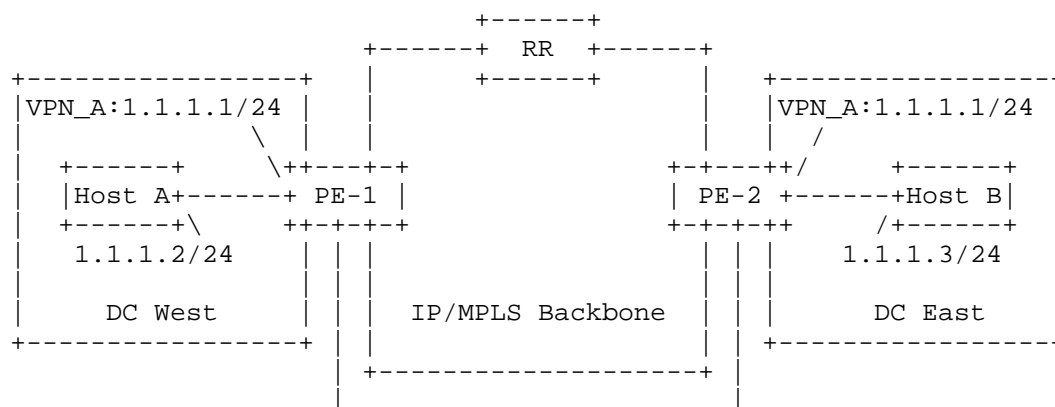
host routes before reflecting them to its clients.

- 2) Those host routes which have been attached with the "can suppress" tag would not be installed into FIBs by clients who are VA-aware since they are not APRs for those host routes. In addition, the RR as an APR would advertise the corresponding VP routes to all of its

clients, and those of which who are VA-aware in turn would install these VP routes into their FIBs.

- 3) Upon receiving a packet from a local CE host, if no matching host route found, the ingress PE router will forward the packet to the RR according to one of the VP routes learnt from the RR, which in turn forwards the packet to the relevant egress PE router according to the host route learnt from that egress PE router. In a word, the FIB table size of PE routers can be greatly reduced at the cost of path stretch. Note that in the case where the RR is not available for transferring L3VPN traffic between PE routers for some reason (e.g., the RR is implemented on a server, rather than a router), the APR function could actually be performed by a given PE router other than the RR as long as that PE router has installed all host routes belonging to the virtual subnet into its FIB. Thus, the RR only needs to attach a "can-suppress" tag to the host routes learnt from its clients before reflecting them to the other clients. Furthermore, PE routers themselves could directly attach the "can-suppress" tag to those host routes for their local CE hosts before distributing them to remote peers as well.
- 4) Provided a given local CE host sends an ARP request for a remote CE host, the PE router that receives such request will install the host route for that remote CE host into its FIB, in case there is a host route for that CE host in its RIB and has not yet been installed into the FIB. Therefore, the subsequent packets destined for that remote CE host will be forwarded directly to the egress PE router. To save the FIB space, FIB entries corresponding to remote host routes which have been attached with "can-suppress" tags would expire if they have not been used for forwarding packets for a certain period of time.

3.6.3. PE Router RIB Reduction



Internet-Draft	Virtual Subnet			July 2013			
VRF_A :	V			VRF_A : V			
	+	+	+	+	+	+	+
1		Prefix	Nexthop Protocol		Prefix	Nexthop Protoco	
	+	+	+	+	+	+	+
		1.1.1.1/32	127.0.0.1 Direct		1.1.1.1/32	127.0.0.1 Direct	
	+	+	+	+	+	+	+
		1.1.1.2/32	1.1.1.2 Direct		1.1.1.3/32	1.1.1.3 Direct	
	+	+	+	+	+	+	+
		1.1.1.0/25	RR IBGP		1.1.1.0/25	RR IBGP	
	+	+	+	+	+	+	+
		1.1.1.128/25	RR IBGP		1.1.1.128/25	RR IBGP	
	+	+	+	+	+	+	+
		1.1.1.0/24	1.1.1.1 Direct		1.1.1.0/24	1.1.1.1 Direct	
	+	+	+	+	+	+	+

Figure 6: RIB Reduction Example

To reduce the RIB size of PE routers, BGP Outbound Route Filtering (ORF) mechanism is used to realize on-demand route announcement. Take the VPN instance A shown in Figure 6 as an example, the procedures of RIB reduction are as follows:

- 1) PE routers as RR clients advertise host routes for their local CE hosts to a RR which however doesn't reflect these host routes by default unless it receives explicit ORF requests for them from its clients. The RR is configured with routes for more specific subnets (e.g., 1.1.1.0/25 and 1.1.1.128/25) corresponding to the virtual subnet (i.e., 1.1.1.0/24) with next-hop being pointed to Null0 and then advertises these routes to its clients via BGP.
- 2) Upon receiving a packet from a local CE host, if no matching host route found, the ingress PE router will forward the packet to the RR according to one of the subnet routes learnt from the RR, which in turn forwards the packet to the relevant egress PE router according to the host route learnt from that egress PE router. In a word, the RIB table size of PE routers can be greatly reduced at the cost of path stretch.
- 3) Just as the approach mentioned in section 3.6.2, in the case where the RR is not available for transferring L3VPN traffic between PE routers for some reason, a PE router other than the RR could advertise the more specific subnet routes as long as that PE router has installed all host routes belonging to that virtual subnet into its FIB.
- 4) Provided a given local CE host sends an ARP request for a remote CE host, the ingress PE router that receives such request will request the corresponding host route from its RR by using the ORF

mechanism (e.g., a group ORF containing Route-Target (RT) and prefix information) in case there is no host route for that CE host in its RIB yet. Once the host route for the remote CE host is

learned from the RR, the subsequent packets destined for that CE host would be forwarded directly to the egress PE router. Note that the RIB entries of remote host routes could expire if they have not been used for forwarding packets for a certain period of time. Once the expiration time for a given RIB entry is approaching, the PE router would notify its RR not to pass the updates for corresponding host route by using the ORF mechanism.

3.7. ARP/ND Cache Table Scalability on Default Gateways

In case where data center default gateway functions are implemented on PE routers of the VS as shown in Figure 4, since the ARP/ND cache table on each PE router only needs to contain ARP/ND entries of local CE hosts, the ARP/ND cache table size will not grow as the number of data centers to be connected increases.

3.8. ARP/ND and Unknown Unicast Flood Avoidance

In VS, the flooding domain associated with a given virtual subnet that has been extended across multiple data centers, has been partitioned into segments and each segment is confined within a single data center. Therefore, the performance impact on networks and servers caused by the flooding of ARP/ND broadcast/multicast and unknown unicast traffic is alleviated.

3.9. Path Optimization

Take the scenario shown in Figure 4 as an example, to optimize the forwarding path for traffic between cloud users and cloud data centers, PE routers located at cloud data centers (i.e., PE-1 and PE-2), which are also data center default gateways, propagate host routes for their local CE hosts respectively to remote PE routers which are attached to cloud user sites (i.e., PE-3).

As such, traffic from cloud user sites to a given server on the virtual subnet which has been extended across data centers would be forwarded directly to the data center location where that server resides, since traffic is now forwarded according to the host route for that server, rather than the subnet route.

Furthermore, for traffic coming from cloud data centers and forwarded to cloud user sites, each PE router acting as a default gateway would forward the traffic received from its local CE hosts according to the best-match route in the corresponding VRF. As a result, traffic from data centers to cloud user sites is forwarded along the optimal path as well.

4. Considerations for Non-IP traffic

Although most traffic within and across data centers is IP traffic, there may still be a few legacy clustering applications which rely on non-IP communications (e.g., heartbeat messages between cluster nodes). To support those few non-IP traffic (if present) in the Virtual Subnet solution, the approach following the idea of "route all IP traffic, bridge non-IP traffic" could be considered as an enhancement to the original Virtual Subnet solution.

Note that more and more cluster vendors are offering clustering applications based on Layer 3 interconnection.

5. Security Considerations

This document doesn't introduce additional security risk to BGP/MPLS L3VPN, nor does it provide any additional security feature for BGP/MPLS L3VPN.

6. IANA Considerations

There is no requirement for any IANA action.

7. Acknowledgements

Thanks to Dino Farinacci, Himanshu Shah, Nabil Bitar, Giles Heron, Ronald Bonica, Monique Morrow, Rajiv Asati and Eric Osborne for their valuable comments and suggestions on this document.

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

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

[MVPN] Rosen. E and Aggarwal. R, "Multicast in MPLS/BGP IP VPNs", draft-ietf-l3vpn-2547bis-mcast-10.txt, Work in Progress, January 2010.

- [VA-AUTO] Francis, P., Xu, X., Ballani, H., Jen, D., Raszuk, R., and L. Zhang, "Auto-Configuration in Virtual Aggregation", draft-ietf-grow-va-auto-05.txt, Work in Progress, December 2011.
- [RFC925] Postel, J., "Multi-LAN Address Resolution", RFC-925, USC Information Sciences Institute, October 1984.
- [RFC1027] Smoot Carl-Mitchell, John S. Quarterman, "Using ARP to Implement Transparent Subnet Gateways", RFC 1027, October 1987.
- [RFC4389] D. Thaler, M. Talwar, and C. Patel, "Neighbor Discovery Proxies (ND Proxy) ", RFC 4389, April 2006.
- [RFC5798] S. Nadas., "Virtual Router Redundancy Protocol", RFC 5798, March 2010.
- [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.
- [802.1AB] IEEE Standard 802.1AB-2009, "Station and Media Access Control Connectivity Discovery", September 17, 2009.
- [802.1Qbg] IEEE Draft Standard P802.1Qbg/D2.0, "Virtual Bridged Local Area Networks -Amendment XX: Edge Virtual Bridging", Work in Progress, December 1, 2011.
- [RFC6820] Narten, T., Karir, M., and I. Foo, "Problem Statement for ARMD", RFC 6820, January 2013.

Authors' Addresses

Xiaohu Xu
Huawei Technologies,
Beijing, China.
Phone: +86 10 60610041
Email: xuxiaohu@huawei.com

Susan Hares
Email: shares@ndzh.com

Internet-Draft

Virtual Subnet

July 2013

Yongbing Fan
Guangzhou Institute, China Telecom
Guangzhou, China.
Phone: +86 20 38639121
Email: fanyb@gsta.com

Christian Jacquenet
France Telecom
Rennes
France
Email: christian.jacquenet@orange.com

Network working group
Internet Draft
Category: Standard Track

L. Yong
X. Xu
Huawei

Expires: April 2014

October 17, 2013

NVGRE and VXLAN Encapsulation Extension for L3 Overlay
draft-yong-l3vpn-nvgre-vxlan-encap-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/ietf/lid-abstracts.txt>.

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

This Internet-Draft will expire on April 17, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

Both NVGRE and VXLAN encapsulations were originally designed for L2 overlay only. This draft proposes the enhancement on both to support L3 overlay as well. The proposed method completely decouples the L3 overlay from the L2 overlay in terms of encoding schema and data processing.

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].

Table of Contents

1. Introduction.....	3
2. NVGRE Encapsulation Extension for L3 Overlay.....	3
3. VXLAN Encapsulation Extension for L3 Overlay.....	3
4. Security Considerations.....	4
5. IANA Considerations.....	5
6. References.....	5
6.1. Normative References.....	5
6.2. Informative References.....	5

1. Introduction

Network Virtualization Overlay [NVO3FRWK] explicitly states that both L2 and L3 overlays are needed in practice. However both NVGRE encapsulation [NVGRE] and VXLAN encapsulation [VXLAN] were originally designed for L2 overlay only.

This document proposes enhancements to NVGRE and VXLAN encapsulations to allow the same data encapsulation semantics for both L2 overlay and L3 overlay. The benefits of this approach are generalizing the data encapsulation semantics for overlay technologies, maintaining L3 overlay natively, and decoupling it from L2 overlay completely.

2. NVGRE Encapsulation Extension for L3 Overlay

NVGER [NVGRE] leverages the GRE protocol [RFC2890] and specifies that the protocol type field in the GRE header MUST be filled with the value of 0x6558, which means for Transparent Ethernet.

This document proposes the protocol type field to be filled with the value of 0x6558, 0x0800(IPv4), or 0x86dd(IPv6). The value of 0x0800 and 0x86dd means that the payload is IP. The value 0x6558 MUST be used if the inner header is an Ethernet header. When NVGRE encapsulation is used for L3 overlay, it MUST use the value of 0x0800 or 0x86dd in the protocol type field and MUST encode an IPv4 or IPv6 header as the inner header. Other fields in the outer header and the GRE header remain the same.

To support backward compatibility, when the remote tunnel end point only support the NVGRE described in [NVGRE], the tunnel end point that supports NVGRE described in this document MUST only encapsulate L2 packets. This capability can be either manually configured or be dynamically informed. How tunnel end points inform each other the encapsulation capabilities is beyond the scope of this document. Note that a tunnel may have more than two end points.

3. VXLAN Encapsulation Extension for L3 Overlay

This document proposes adding a protocol type field in the VXLAN header as shown below. It takes 16 bits from the reserved 24 bits as the protocol type field. The remained 8 reserved bits MUST be filled with zero. For L2 overlay encapsulation, the protocol type field MUST be filled with the value of 0x6558 and inner header MUST be an Ethernet header. For L3 overlay encapsulation, the protocol type

field MUST be filled with the value of 0x0800(IPv4) or 0x86dd(IPv6), and inner header MUST be an IPv4 or IPv6 header. Other fields in the outer header and VXLAN header remain the same.

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Outer Ethernet Header:
      As described in VXLAN [VXLAN]
Outer IP Header:
      As described in VXLAN [VXLAN]
Outer UDP Header:
      As described in VXLAN [VXLAN]
VXLAN Header:
+++++
|R|R|R|R|I|R|R|R|   Reserved   |Prot. Type=0x6558/0x0800/0x86dd|
+++++
|               VXLAN Network Identifier (VNI) |   Reserved   |
+++++
Inner Header:
+++++
|               Ethernet header or IP Header               ~
+++++

```

To be backward compatible with the existing VXLAN encapsulation [VXLAN], the value 0x0000 in the Protocol Type field MUST be treated as Ethernet payload too. When the end points of a tunnel support different VXLAN formats, i.e. one, say A, supports old VXLAN format and another, say B, supports the new format described in this document, B MUST only encapsulate L2 packets and set value 0x0000 in the protocol type field. This capability can be either manually configured at B or be dynamically informed. How tunnel end points inform each other the encapsulation capabilities is beyond the scope of this document. Note that a tunnel may have more than two end points.

Having protocol type field in the VXLAN header enables other overlay payload type beside L2 and L3 overlays. The application for other payload type is for future study.

4. Security Considerations

The mechanism proposed in this document does not add any additional security concern beside what has been described in the NVGRE [NVGRE] and VXLAN [VXLAN].

5. IANA Considerations

The document does not require any IANA action.

6. References

6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC2119, March 1997.
- [RFC2890] Dommety, G., "Key and Sequence Number Extension to GRE", RFC2890, September 2000

6.2. Informative References

- [NVO3FRWK] Lasserre, M., et al, "Framework for DC Network Virtualization", draft-ietf-nvo3-framework-03.txt, work in progress.
- [NVGRE] Sridharan, M., et al, "NVGRE: Network Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre-03, work in progress
- [VXLAN] Mahalingam, M., Dutt, D., etc, "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", draft-mahalingam-dutt-dcops-vxlan-05.txt, work in progress

Authors' Addresses

Lucy Yong
Huawei Technologies, USA

Phone: 918-808-1918
Email: lucy.yong@huawei.com

Xiaohu Xu
Huawei Technologies,
Beijing, China

Phone: +86-10-60610041
Email: xuxiaohu@huawei.com

L3 VPN Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 11, 2013

J. Zhang
Juniper Networks, Inc.
January 07, 2013

L2L3 VPN Multicast MIB
draft-zzhang-l2l3-vpn-mcast-mib-00

Abstract

This memo defines an experimental portion of the Management Information Base for use with network management protocols in the Internet community.

In particular, it describes managed objects common to both VPLS and VPN Multicast.

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 July 11, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. The Internet-Standard Management Framework	3
3. Conventions	3
4. Summary of MIB Module	3
5. Definitions	3
6. Security Considerations	8
7. IANA Considerations	8
8. References	8
8.1. Normative References	8
8.2. Informative References	9

1. Introduction

Multicast in VPLS and VPN can be achieved by using provider tunnels to deliver to all or a subset of PEs. The signaling of provider tunnel choice is very similar for both VPLS and VPN multicast (aka MVPN), and this memo describes managed objects common to both VPLS Multicast [I-D.ietf-l2vpn-vpls-mcast] and MVPN [RFC 6513/6514].

2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

3. 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 [RFC2119].

4. Summary of MIB Module

L2L3-VPN-MCAST-MIB contains a Textual Convention, L2L3VpnMcastProviderTunnelType, and a L2L3VpnMcastPmsiTunnelAttributeTable. Other MIB objects ([I-D.ietf-l2vpn-vpls-mcast], [I-D.ietf-l3vpn-mvpn-mib]) may point to entries in the L2L3VpnMcastPmsiTunnelAttributeTable.

5. Definitions

```
L2L3-VPN-MCAST-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,  
    experimental, Unsigned32  
    FROM SNMPv2-SMI
```

```
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
```



```
FROM SNMPv2-CONF

TruthValue, RowPointer, RowStatus, TimeStamp, TimeInterval
FROM SNMPv2-TC

SnmAdminString
FROM SNMP-FRAMEWORK-MIB

InetAddress, InetAddressType
FROM INET-ADDRESS-MIB

MplsLabel
FROM MPLS-TC-STD-MIB

l2L3VpnMcastMIB MODULE-IDENTITY
    LAST-UPDATED "201301071200Z" -- 07 January 2013 12:00:00 GMT
    ORGANIZATION "IETF Layer-3 Virtual Private
        Networks Working Group."
    CONTACT-INFO

        "
        Comments and discussion to l3vpn@ietf.org
        Jeffrey (Zhaohui) Zhang
        Juniper Networks, Inc.
        10 Technology Park Drive
        Westford, MA 01886
        USA
        Email: zzhang@juniper.net
        "

    DESCRIPTION
        "This MIB contains common managed object definitions for
        multicast in Layer 2 and Layer 3 VPNs, defined by
        [I-D.ietf-l2vpn-vpls-mcast] and RFC 6513/6514.
        Copyright (C) The Internet Society (2013)."
```

-- Revision history.

```
REVISION "201301071200Z" -- 07 January 2013 12:00:00 GMT
DESCRIPTION
    "Initial version of the draft."
 ::= { experimental 99 } -- number to be assigned

-- Textual convention

l2L3VpnMcastProviderTunnelType ::= TEXTUAL-CONVENTION
    SYNTAX      INTEGER { unconfigured (0),
                          pim-asm (1),
```

```

        pim-ssm (2),
        pim-bidir (3),
        rsvp-p2mp (4),
        ldp-p2mp (5),
        ingress-replication (6),
        ldp-mp2mp (7)
    }
    STATUS          current
    DESCRIPTION
        "Types of provider tunnels used for multicast in a l2/l3vpn."
    REFERENCE
        "[RFC6514]"

-- Top level components of this MIB.
-- tables, scalars

l2L3VpnMcastObjects OBJECT IDENTIFIER ::= { l2L3VpnMcastMIB 1 }
l2L3VpnMcastStates  OBJECT IDENTIFIER ::= { l2L3VpnMcastObjects 1 }

-- Table of PMSI attributes

l2L3VpnMcastPmsiTunnelAttributeTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF L2L3VpnMcastPmsiTunnelAttributeEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table is for advertised/received PMSI attributes,
         to be referred to by I-PMSI or S-PMSI table entries"
        ::= {l2L3VpnMcastStates 1 }

l2L3VpnMcastPmsiTunnelAttributeEntry OBJECT-TYPE
    SYNTAX          L2L3VpnMcastPmsiTunnelAttributeEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table corresponds to an PMSI attribute
         that is advertised/received on this router.
         For BGP-based signaling (for I-PMSI via auto-discovery
         procedure, or for S-PMSI via S-PMSI A-D routes),
         they are just as signaled by BGP (RFC 6514 section 5,
         'PMSI Tunnel attribute').
         For UDP-based S-PMSI signaling for PIM-MVPN,
         they're derived from S-PMSI Join Message
         (RFC 6513 section 7.4.2, 'UDP-based Protocol')..

         Note that BGP-based signaling may be used for
         PIM-MVPN as well."
    INDEX {

```

```

        12L3VpnMcastPmsiTunnelAttributeFlags,
        12L3VpnMcastPmsiTunnelAttributeType,
        12L3VpnMcastPmsiTunnelAttributeLabel,
        12L3VpnMcastPmsiTunnelAttributeId
    }
 ::= { 12L3VpnMcastPmsiTunnelAttributeTable 1 }

L2L3VpnMcastPmsiTunnelAttributeEntry ::= SEQUENCE {
    12L3VpnMcastPmsiTunnelAttributeFlags    OCTET STRING,
    12L3VpnMcastPmsiTunnelAttributeType      Unsigned32,
    12L3VpnMcastPmsiTunnelAttributeLabel     MplsLabel,
    12L3VpnMcastPmsiTunnelAttributeId        OCTET STRING,
    12L3VpnMcastPmsiTunnelPointer            RowPointer,
    12L3VpnMcastPmsiTunnelIf                 RowPointer
}

12L3VpnMcastPmsiTunnelAttributeFlags OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (1))
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "For UDP-based S-PMSI signaling for PIM-MVPN, this is 0.
        For BGP-based I/S-PMSI signaling,
        per RFC 6514 section 5, 'PMSI Tunnel Attribute':

The Flags field has the following format:

```

```

    0 1 2 3 4 5 6 7
    +---+---+---+---+
    | reserved |L|
    +---+---+---+---+

```

This document defines the following flags:

```

    + Leaf Information Required (L)"
 ::= { 12L3VpnMcastPmsiTunnelAttributeEntry 1 }

12L3VpnMcastPmsiTunnelAttributeType OBJECT-TYPE
    SYNTAX      L2L3VpnMcastProviderTunnelType
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "For BGP-based I/S-PMSI signaling for either PIM or BGP-MVPN,
        per RFC 6514 section 5, 'PMSI Tunnel Attribute':

```

The Tunnel Type identifies the type of the tunneling technology used to establish the PMSI tunnel. The type determines the syntax and semantics of the Tunnel Identifier field. This document defines the

following Tunnel Types:

- 0 - No tunnel information present
- 1 - RSVP-TE P2MP LSP
- 2 - mLDP P2MP LSP
- 3 - PIM-SSM Tree
- 4 - PIM-SM Tree
- 5 - PIM-Bidir Tree
- 6 - Ingress Replication
- 7 - mLDP MP2MP LSP

For UDP-based S-PMSI signaling for PIM-MVPN, RFC 6513 does not specify if a PIM provider tunnel is SSM, SM or Bidir, and an agent can use either type 3, 4, or 5 based on its best knowledge."

```
::= { 12L3VpnMcastPmsiTunnelAttributeEntry 2 }
```

12L3VpnMcastPmsiTunnelAttributeLabel OBJECT-TYPE

```
SYNTAX          MplsLabel
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
```

```
"For BGP-based I/S-PMSI signaling,
per RFC 6514 section 5, 'PMSI Tunnel Attribute':
```

If the MPLS Label field is non-zero, then it contains an MPLS label encoded as 3 octets, where the high-order 20 bits contain the label value. Absence of MPLS Label is indicated by setting the MPLS Label field to zero.

For UDP-based S-PMSI signaling for PIM-MVPN, this is not applicable for now, as RFC 6513 does not specify mpls encapsulation and tunnel aggregation with UDP-based signaling."

```
::= { 12L3VpnMcastPmsiTunnelAttributeEntry 3 }
```

12L3VpnMcastPmsiTunnelAttributeId OBJECT-TYPE

```
SYNTAX          OCTET STRING ( SIZE (4|8|12) )
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
```

```
"For BGP-based signaling, as defined in RFC 6514 section 5,
'PMSI Tunnel Attribute'.
```

For UDP-based S-PMSI signaling for PIM-MVPN, RFC 6513 only specifies the 'P-Group' address, and that is filled into the first four octets of this field."

```
::= { 12L3VpnMcastPmsiTunnelAttributeEntry 4 }
```

l2L3VpnMcastPmsiTunnelPointer OBJECT-TYPE

SYNTAX RowPointer

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If the tunnel exists in some MIB table, this is the row pointer to it."

::= { l2L3VpnMcastPmsiTunnelAttributeEntry 5 }

l2L3VpnMcastPmsiTunnelIf OBJECT-TYPE

SYNTAX RowPointer

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If the tunnel has a corresponding interface, this is the row pointer to the ifName table."

::= { l2L3VpnMcastPmsiTunnelAttributeEntry 6 }

END

6. Security Considerations

N/A

7. IANA Considerations

IANA is requested to root MIB objects in the MIB module contained in this document under the transmission subtree.

.

8. References

8.1. Normative References

- [RFC3418] Presuhn, R., "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3418, December 2002.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Ed., Perkins, D., Ed.,

and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

- [RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.
- [RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.
- [I-D.ietf-l2vpn-vpls-mcast] Aggarwal, R., Rekhter, Y., Kamite, Y., and L. Fang, "Multicast in VPLS", draft-ietf-l2vpn-vpls-mcast-11 (work in progress), July 2012.
- [I-D.ietf-l2vpn-vpls-mib] Nadeau, T., Koushik, K., and R. Mediratta, "Virtual Private Lan Services (VPLS) Management Information Base", draft-ietf-l2vpn-vpls-mib-07 (work in progress), September 2012.

8.2. Informative References

- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

Author's Address

Zhaohui Zhang
Juniper Networks, Inc.
10 Technology Park Drive
Westford, MA 01886
USA

EMail: zzhang@juniper.net

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 20, 2013

Zhang
Giuliano
Juniper Networks
Pacella
Verizon
February 16, 2013

Global Table Multicast with BGP-MVPN Procedures
draft-zzhang-mboned-mvpn-global-table-mcast-00.txt

Abstract

This document describes a way to implement Global Table Multicast, aka Internet Multicast, using BGP encodings and procedures for MVPN as specified in [RFC6514].

No protocol modification/extension is required. This is purely for informational and clarifying purposes only.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <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 August 20, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

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

Table of Contents

1. Introduction	3
2. Requirements Language	4
3. Operation	5
3.1. IBGP session between BRs and non-BRs	5
3.2. Non-BGP RPF Routes or BGP RPF routes not originated by the BRs	5
4. Security Considerations	8
5. IANA Considerations	9
6. Acknowledgements	10
7. References	11
7.1. Normative References	11
7.2. Informative References	11
Authors' Addresses	12

1. Introduction

[RFC6513] and [RFC6514] specify procedures and encodings to implement Multicast for L3VPNs (MVPN). [RFC6513] specifies general concepts and procedures that apply to PIM-based and/or BGP-based C-Multicast State Signaling (referred to PIM-MVPN and BGP-MVPN respectively), and [RFC6514] specifies BGP procedures and encodings used by both PIM-MVPN and BGP-MVPN.

While [RFC6513] and [RFC6514] assume the context of VPN, they can be used to implement Global (vs. VRF) Table Multicast as well, without any protocol modification/extension, even though the RFCs do not explicitly mention it.

Consider a provider network where the "core" part of it uses MPLS P2MP LSPs or Ingress Replication over either P2P LSPs (with RSVP-TE) or MP2P LSPs (with LDP) instead of PIM to carry multicast traffic among the border routers (BRs) of the core. Those BRs run PIM on interfaces connected to other routers outside the core.

This document describes how Global Table Multicast can be implemented using BGP-MVPN procedures. We start with a simple reference scenario below, and also discuss one slightly different scenario and another special one.

With Global Table Multicast implemented by BGP-MVPN procedures, all the features/characteristics of BGP-MVPN apply, including scaling, aggregation, flexible choice of provider tunnels, support for PIM-DM/ASM/SSM/Bidir as PE-CE multicast protocol, BSR and AUTO-RP as RP-to-group mapping protocols, etc.

2. Requirements Language

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

3. Operation

In the simplest reference scenario, the BRs advertise to each other RPF routes to multicast sources via iBGP (with or without RRs in the middle) with Next Hop set to themselves. The routes could have been learned from other non-BRs via eBGP or IGP.

Conceptually and functionally, those BRs are just like MVPN PEs: connections to other routers outside the core can be treated as PE-CE interfaces and MVPN procedures can run among the PEs (i.e., BRs) for Discovery, Tunnel Binding, and Multicast State Signaling.

With that, using BGP-MVPN procedures for Global Table Multicast is straightforward and requires almost no further clarification. However, some popular practices are described below.

By default, RD 0:0 is used when advertising A-D routes for Global Table Multicast, though an implementation MAY support the configuration and use of a different RD value.

Similarly, when constructing the C-multicast Import RT as specified in Section 7 of [RFC6514], it is RECOMMENDED that the Local Administrator field is set to 0, though an implementation MAY use any value that can uniquely associate it to the global routing table (vs. a VRF).

3.1. IBGP session between BRs and non-BRs

In the simple reference scenario above, it is assumed that the BRs learn RPF routes from non-BRs via eBGP or IGP. The assumption is to illustrate the analogy to a true VPN environment. In another deployment scenario, the BRs could have learned the RPF routes over those iBGP sessions to non-BRs. If the BRs act as RRs and reflect the RPF routes to other BRs with policies to attach VRF Route Import Extended Community and Source AS Extended Community, BGP-MVPN procedures can still be used as described earlier. Even if the BRs do not act as RRs, the scenario could be considered analogous to what [RFC6368] describes. As long as BRs re-advertise those RPF routes with the above mentioned communities, BGP-MVPN procedures can be used as described earlier. Note that they do not even need to use the push/pop procedures in [RFC6368] - the only requirement is for the BRs to re-advertise the routes learned over iBGP sessions from non-BRs to other BRs over iBGP sessions.

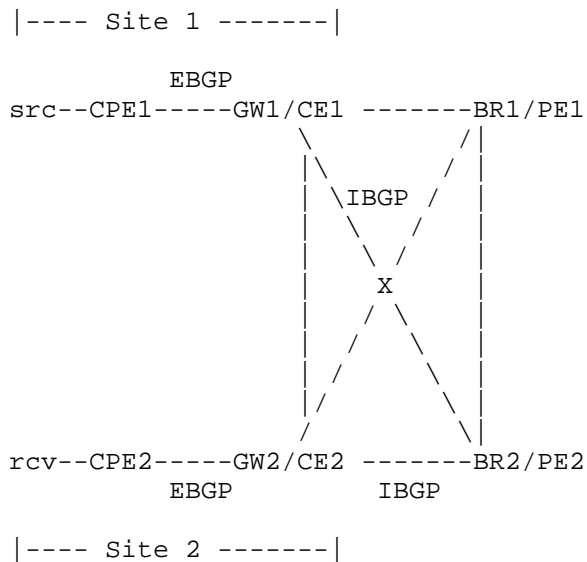
3.2. Non-BGP RPF Routes or BGP RPF routes not originated by the BRs

With true MVPN, the PEs advertise the RPF routes themselves as VPN-IP routes, and attach a VRF Route Import Extended Community that has the

C-multicast Import RT value for the VRF associated with the routes. The VRF Route Import Extended Community is extracted by egress PEs and attached to their C-Multicast Routes as Route Target Extended Community to control the distribution to and importation by relevant ingress PEs.

With Global Table Multicast, in both the simple reference scenario and the above mentioned variance, the BRs do (re-)advertise the RPF routes as required for BGP-MVPN. However, in other situations, it is possible that the RPF routes are not advertised by the BRs via BGP at all, hence they may not carry the VRF Route Import Extended Community.

Consider the following example:



There is a full-mesh of IBGP sessions among provider routers GW1/BR1/BR2/GW2. EBGP sessions run between CPE1/GW1 and between CPE2/GW2. Border routers BR1/BR2 run BGP-MVPN procedures for Global Table Multicast. GW1 learns of BGP route to the src from CPE1 and advertises it to BRx/GW2.

Because GW1 does not run MVPN, BR2's route to the src (learned from GW1 instead of BR1) does not have the VRF Route Import Extended Community. Therefore, it would not be able to correctly attach a Route Target Extended Community corresponding to BR1 in its C-Multicast Routes.

To handle that situation, BR2 performs the following recursively. Note that the route in the following procedure is either the RPF route for the source itself, or the route to the Next Hop of the BGP route in the previous recursion.

- o If the route is a BGP route with a VRF Route Import Extended Community, that VRF Route Import Extended Community is used.
- o If the route is a BGP route without a VRF Route Import Extended Community, get the route to the Next Hop and recurse.
- o If the route is an IGP route with a RSVP-TE LSP as next hop, and the LSP endpoint is a BR that advertises an Intra-AS I-PMSI A-D route (BR1 in the above example), a VRF Route Import Extended Community is constructed as BR_addr:0 to be associated with the RPF route, where the BR_addr is the Originating Router's IP Address of the Intra-AS I-PMSI A-D route.

If the above procedure does not produce a usable VRF Route Import Extended Community, then the RPF route is considered a local route (vs. a remote route that is associated with a remote BR). Note that the special process is necessary only if the BRs (that run MVPN procedures) do not advertise the RPF routes via BGP and include VRF Route Import Extended Community in such routes.

Constructing the VRF Route Import as BR_addr:0 by an egress BR in the above special situation explains why it is RECOMMENDED that the Local Administrator is set to 0 when an ingress BR constructs its C-Multicast Import RT - the zero value is a special value agreed on apriori by all (vs. a local value that is normally picked by the ingress router and signaled via the VRF Route Import Extended Community).

4. Security Considerations

This document raises no new security issues. Security considerations for the base protocol are covered in [RFC6514].

5. IANA Considerations

This document has no IANA considerations.

This section should be removed by the RFC Editor prior to final publication.

6. Acknowledgements

The authors would like to thank Rahul Aggarwal and Yakov Rehkter for their comments and suggestions.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.
- [RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.

7.2. Informative References

- [RFC6368] Marques, P., Raszuk, R., Patel, K., Kumaki, K., and T. Yamagata, "Internal BGP as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 6368, September 2011.

Authors' Addresses

Jeffrey Zhang
Juniper Networks
10 Technology Park Dr.
Westford, MA 01886
US

Email: zzhang@juniper.net

Lenny Giuliano
Juniper Networks
2251 Corporate Park Drive
Herndon, VA 20171
US

Email: lenny@juniper.net

Dante J. Pacella
Verizon Communications
22001 Loudoun County Parkway
Ashburn, VA 20147

Email: dante.j.pacella@verizonbusiness.com

