OpenFlow enabled Integrated Routing and Bridging over Ethernet Virtual Private Networks

Panagiotis Karamichailidis, Kostas Choumas and Thanasis Korakis
Dept. of ECE, University of Thessaly, Volos, Greece
Email: {karamiha, kohoumas, korakis}@uth.gr

***Abstract*—Ethernet Virtual Private Network is an emerging Data Center Interconnect technology, which relies on Multiprotocol BGP and an encapsulation protocol over IP, such as Virtual Extensible LAN. It enables the creation of Layer 2 overlays on top of provider IP networks, which can be interconnected through the symmetric or asymmetric Integrated Routing and Bridging extension. In this work, we present our testbed implementation of these technologies using OpenFlow and we compare the advantages and disadvantages of the symmetric and asymmetric solutions.**

I. Introduction

Virtualization is essentially the abstraction of physical resources to support multitenancy, allowing for fully isolated and distributed environments. Network virtualization provides abstraction for network resources and allows the creation of virtual networks as network overlays. Data centers ensued from the server virtualization require their integration with the rich results of the network virtualization, since network services have to be distributed in multiple data centers for geographical diversity and failure resilience. Data Center Interconnect (DCI) technologies have proliferated in recent years and support network overlays enabling multiple data centers to work together.

In many scenarios, DCI has to support Layer 2 overlays, which are stretched broadcast domains across multiple data centers, enabling e.g. the Virtual Machine (VM) migration from one data center to the other. Ethernet Virtual Private Network (**EVPN**) [1] is an emerging DCI solution [2] supporting Layer 2 overlays. EVPN explores Multiprotocol Border Gateway Protocol (**MP-BGP**) for the control plane and an encapsulation protocol over IP for the data plane tunneling [3], such as Virtual Extensible LAN (**VXLAN**) [4]. Each overlay is logically equivalent to an **IP subnet**, which needs routing for being connected to other subnets.

Integrated Routing and Bridging (**IRB**) [5] enables inter-subnet communication without the use of external routers. IRB can be either **symmetric** or **asymmetric**. Symmetric IRB seems to be more scalable, however, it requires additional tunnels and results in higher latency than asymmetric IRB. In this paper, an existing **OpenFlow** implementation for VXLAN based EVPN is extended to support the two IRB solutions. To the best of our knowledge, this is the first study comparing the two solutions.

Next, Section II briefly summarizes the EVPN and IRB abstraction. In section III, we explain the details on the OpenFlow implementation of these technologies, while in Section IV we present an evaluation of the two IRB solutions. Section V conludes the paper.

978-1-7281-4973-820$31.00

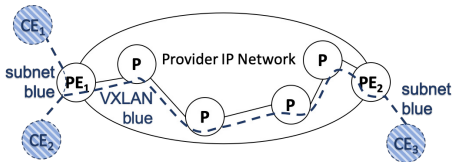


Fig. 1: VLAN-based EVPN instance using VXLAN

II. EVPN and IRB

A. EVPN for intra-subnet communication

The EVPN technology decouples the data plane from the control plane. Without loss of generality, we focus on the VLAN-based operation of EVPN, where each EVPN instance consists of a single subnet (a.k.a. Layer 2 overlay), leaving VLAN-aware and VLAN-bundle operations for future work. Each subnet comprises Customer Edge devices (**CE**s) of the same customer attached to the Provider Edge devices (**PE**s) of a provider network, which are interconnected through the Provider devices (**P**s), as shown in Figure 1. The CEs are the end points of the customer traffic, including physical devices and VMs in data centers. The provider network may be an MPLS or IP network, which provides virtual Layer 2 connectivity (a.k.a. bridging) between the CEs. In case of IP provider networks, there are multiple options for the IP tunneling used for the bridging, such as the VXLAN technology, that is the focus of this paper.

1) VXLAN as data plane: VXLAN is a VLAN-like tunneling technique to encapsulate Layer 2 Ethernet frames withing Layer 4 UDP datagrams. The endpoints of the VXLAN tunnels may be either virtual or physical switch ports, which are known as VXLAN Tunnel End-Points (VTEPs). Each tunnel is identified by a Virtual Network Identifier (**VNI**), which identifies the subnet. Same subnet CEs, attached to different PEs, are bridged with the assistance of a VXLAN tunnel. PEs are equipped with multiple MAC Virtual Routing and Forwarding (**MAC-VRF**s) tables [3], each one corresponding to a different EVPN instance and including one Bridge Table (**BT**) per subnet of this instance. In our case of VLAN-based EVPN operation, there is an **one-to-one mapping between MAC-VRF and BT**.

Each BT is actually located in a PE connecting the CEs of a specific subnet. It behaves as a learning-switch for the local CEs attached to this PE (e.g. CE_1 and CE_2 of Figure 1) and forwards to remote CEs attached to other PEs through

(a) Symmetric IRB

(b) Asymmetric IRB

Fig. 2: IRB solution types for VXLAN based EVPN

VXLAN tunnels (e.g. CE_2 and CE_3). When a packet of a subnet is to be forwarded from one PE to another, it is pushed to the corresponding VXLAN tunnel connecting the two PEs and serving the packet's subnet. The forwarding rules for the bridging of the collocated CEs are updated according to the switch's learning process, while the corresponding rules for the remote CEs are controlled by MP-BGP.

2) MP-BGP as control plane: According to MP-BGP, the PEs act as BGP peers exchanging EVPN Network Layer Reachability Information (NLRI). The exchanged EVPN NLRI enables each PE to learn the IP and MAC addresses of the remote CEs, which belong to the same subnet with at least one of its local CEs. Thus, the PEs are able to reply to the ARP requests for the IP addresses of the remote CEs, without flooding these requests to the provider network. In this way, EVPN has an important feature comparing to its predecessor VPLS, named *ARP suppression*, which refers to the reduced flooding of ARP request broadcasts.

B. IRB for inter-subnet communication

There are scenarios requiring inter-subnet connectivity between the CEs, apart from the intra-subnet one. The inter-subnet communication is traditionally achieved at centralized routers. In this case, two adjacent CEs of different subnets are able to communicate by backhauling their traffic from their PE to the centralized router and then back to the PE. For today's large-scale data centers, this scheme is very inefficient and sometimes impractical. Integrated Routing and Bridging (IRB) is needed on the PEs to avoid inefficient backhauling.

IRB is implemented by equipping each PE with extra tables, called **IP-VRFs** [5], one for each customer. All BTs corresponding to same customer subnets are connected through specific interfaces, named IRB interfaces, to the same IP-VRF. The IP-VRF routing complements BT bridging. The functionality of the IP-VRF depends on the IRB solution, namely symmetric or asymmetric. In symmetric IRB, as its name implies, the IP-VRF lookup operation is symmetric at both ingress and egress PEs, while in asymmetric, it is completed at the ingress PE.

1) Symmetric IRB: In symmetric IRB, as shown in Figure 2(a), a packet forwarded from CE_1 of the "blue" subnet to CE_2 of the "orange" subnet is firstly pushed at the PE_1's blue BT. The packet is destined to another subnet, thus its MAC destination is this of the gateway connecting the two subnets. More specifically, it is the MAC address of the IRB interface connecting PE_1's blue BT to the IP-VRF of the customer owning these subnets. PE_1's IP-VRF routes the packet to PE_2's IP-VRF, since it knows that CE_2 is attached there. The tunnel that is used for the packet forwarding from PE_1 to PE_2 corresponds to a "neutral" subnet, the "black" one in our case, which is only used for the PEs interconnection. The MAC destination of the packet changes each time it passes through a router, being the CE_2's MAC address when it passes PE_2's IP-VRF. Finally, the packet is forwarded to PE_2's orange BT and then to CE_2.

2) Asymmetric IRB: On the other hand, in asymmetric IRB of Figure 2(b), the same packet is pushed to the PE_1's orange BT, after passing through the PE_1's blue BT and IP-VRF. This requires from PE_1 having a BT even for a subnet that none of its CEs belongs to, e.g. the orange subnet. After the packet being pushed to the orange BT with MAC destination the one of CE_2, it is forwarded in the same way with the intra-subnet communication, as it would happen if CE_1 was belonging to the orange subnet. Thus, the orange tunnel is used for the packet forwarding from PE_1 to PE_2.

III. OpenFlow Implementation

In this paper, we exploit OpenFlow version 1.3 and its NXM (Nicira eXtended Match) extension for implementing EVPN and IRB. We assume that each PE is an OpenFlow switch that is controlled by the Ryu [6] applications *SimpleSwitch13* and *RestVtep*, as well as our developed Ryu application, named *RestIrb*[(1)]. The OpenFlow network of the NITOS testbed [7] is used for our experimentation. NITOS nodes use OvS [8] and Mininet [9] to behave as PEs (virtual OpenFlow switches) with multiple attached CEs (Mininet hosts) and VTEPs interconnecting them through VXLAN tunnels.

Each PE, as an OpenFlow switch, is equipped with a pipeline of OpenFlow tables, including sets of OpenFlow entries handling specific packets (entry's matching criteria) in a specific way (entry's actions). Every new packet arrival, the switch searches the tables, one after the other and always starting from table 0, looking for the entry matching this packet, in order to apply the corresponding actions. Table I presents the OpenFlow entries of every PE. The OpenFlow entries are grouped according to their functionality and each line below the header corresponds to a group. The first column, named "table", indicates the OpenFlow table that each group of entries is included. The next three columns, named "match", "actions" and "#", describe the matching criteria and the

(1)https://www.dropbox.com/s/ycn0enr61sar8ho/rest_irb.py?dl=0

TABLE I: The OpenFlow tables in a PE of an EVPN network using either symmetric or asymmetric IRB.

table	symmetric IRB			asymmetric IRB			group
	match $\Longrightarrow$	**actions**	**#**	**match** $\Longrightarrow$	**actions**	**#**	
0	in_port=port_of_loc_CE, dl_scr=loc_CE	write_meta(VNI), go_to(1)	$\sum_s h_{ps}$	in_port=port_of_loc_CE, dl_scr=loc_CE	write_meta(VNI), go_to(1)	$\sum_s h_{ps}$	1
	in_port=VTEP_to_rem_PE	write_meta(VNI), go_to(1)	$\sum_{p',s} t^s_{pp's} + P - 1$	in_port=VTEP_to_rem_PE	write_meta(VNI), go_to(1)	$\sum_{p',s} t^a_{pp's}$	2
1	metadata=VNI, dl_dst=loc_CE	output(port_of_loc_CE)	$\sum_s h_{ps}$	metadata=VNI, dl_dst=loc_CE	output(port_of_loc_CE)	$\sum_s h_{ps}$	3
	metadata=VNI, dl_dst=rem_CE	output(VTEP_to_rem_PE)	$\sum_{p',s} t^s_{pp's} h_{p's}$	metadata=VNI, dl_dst=rem_CE	output(VTEP_to_rem_PE)	$\sum_{p'\neq p,s} h_{p's}$	4
	metadata=VNI, eth_type=ARP, arp_op=REQ, arp_tpa=CE	create ARP Reply	$\sum_s h_{ps} + \sum_{p',s} t^s_{pp's} h_{p's}$	metadata=VNI, eth_type=ARP, arp_op=REQ, arp_tpa=CE	create ARP Reply	$\sum_{p',s} h_{p's}$	5
	metadata=VNI, dl_dst=loc_IRB	go_to(2)	S	metadata=VNI, dl_dst=loc_IRB	go_to(2)	S	6
	metadata=VNI, eth_type=ARP, arp_op=REQ, arp_tpa=loc_IRB	create ARP Reply	S	metadata=VNI, eth_type=ARP, arp_op=REQ, arp_tpa=loc_IRB	create ARP Reply	S	7
	metadata=neutral-VNI, dl_dst=IRB	output(VTEP_to_rem_PE), if rem_IRB go_to(2), if loc_IRB	P	-	-	0	8
2	eth_type=IP, nw_dst=rem_CE	set_dl_src(IRB), set_dl_dst(rem_IRB), write_meta(neutral-VNI), resubmit(1)	$\sum_{p'\neq p,s} h_{p's}$	eth_type=IP, nw_dst=CE	set_dl_src(IRB), set_dl_dst(CE), write_meta(VNI), resubmit(1)	$\sum_{p',s} h_{p's}$	9
	eth_type=IP, nw_dst=loc_CE	set_dl_src(IRB), set_dl_dst(loc_CE), write_meta(VNI), resubmit(1)	$\sum_s h_{ps}$	-	-	0	10

actions of the entries of each group, as well as the number of the entries included in this group, when "symmetric IRB" is used. Finally, the following three same titled columns give the same information, in case of "asymmetric IRB".

The entries of table 0 are responsible for the slicing between the subnets, while the entries of tables 1 and 2 are for the bridging and routing functionalities respectively. In more detail, the first two groups in table 0 tag the incoming packets from a subnet with the respective VNI. Either if these packets come from a local CE (1st group) or from another PE through a tunnel (2nd group), the switch writes the packet's VNI to the packet's metadata (annotation space that can be read from other OpenFlow entries) and pushes it to table 1. This functionality is the same for both IRB solutions.

In table 1, all entries match at least the packet's VNI for detecting its subnet. Apart from the VNI, the first two groups match the packet's destination MAC address and forward either to a local CE of the same subnet (3rd group) or to a remote CE through the appropriate tunnel (4th group). The 5th group implements the ARP suppression of EVPN by replying to the ARP requests destined to remote CEs, avoiding the ARP flooding into the network. The 6th group matches the packets destined to an IRB interface and pushes them to table 2 for routing, while the 7th group replies to the ARP requests for the IP addresses of the IRB interfaces. Apart from these groups applying to both symmetric and asymmetric IRB, the symmetric one has also the 8th group, which is responsible for the packet forwarding to other PEs through the neutral tunnel or to table 2 for the packets coming from other PEs.

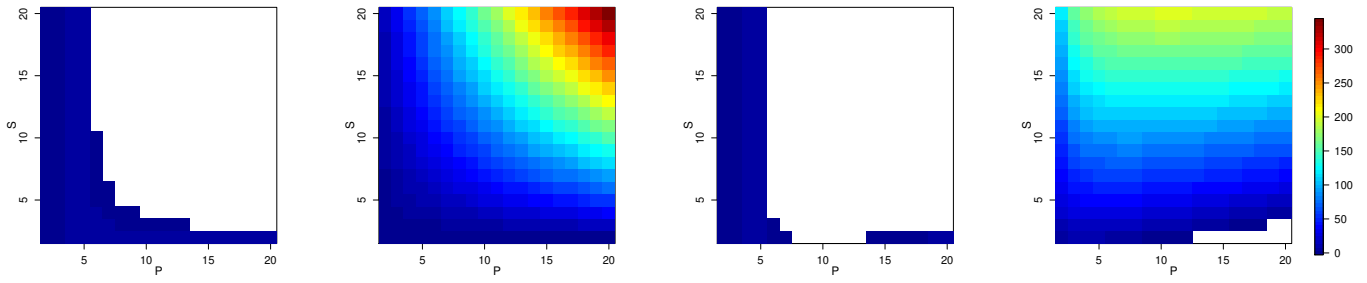
Table 2 implements the routing functionality, thus its entries match the destination IP address. The 9th group applies to both symmetric and asymmetric IRB. In case of symmetric IRB, it matches the packets destined to remote CEs and changes their Ethernet header and subnet. The new destination MAC is this of the remote IRB interface and the new subnet is the neutral one. On the other hand, in case of asymmetric IRB, the new destination MAC is this of the CE, either local or remote, and the new subnet is the one of this CE. Then, the packet is pushed back to table 1 for bridging. Finally, in symmetric IRB, the second lookup happened at the egress PE is operated by the 10th group, which matches the packets destined to local CEs coming from other PEs.

Going back to the examples of Section II-B, the IP packet of Figure 2(a) goes through the OpenFlow entries of groups $1 \rightarrow 6 \rightarrow 9 \rightarrow 8$ at the ingress PE_1. The VNI written in the packet's metadata is this of the blue subnet, after group 1 entry, and then it is changed to the black VNI by group 9 entry. At the egress PE_2, the packet goes through entries of groups $2 \rightarrow 8 \rightarrow 10 \rightarrow 3$, where the packets is mapped to the black VNI by group 2 and then to the orange VNI by group 10. On the other hand, the IP packet of Figure 2(b) goes through the OpenFlow entries of groups $1 \rightarrow 6 \rightarrow 9 \rightarrow 4$ at the ingress PE_1, where the VNI written by group 9 is the orange one (not the black one as before). At the egress PE_2, the packet goes through entries of groups $2 \rightarrow 3$.

IV. Symmetric and Asymmetric IRB Comparison

Let's assume an EVPN network with a set $\mathcal{P}$ of P PEs and a set $\mathcal{S}$ of S subnets, excluding the neutral subnet in case of symmetric IRB. h_{ps} is the number of CEs attached to PE $p \in \mathcal{P}$ and belong to subnet $s \in \mathcal{S}$. Binary $t^s_{pp's} = 1$ if and only if $p \neq p'$, $h_{ps} > 0$ and $h_{p's} > 0$, while $t^a_{pp's} = 1$ if and only if $p \neq p'$ and either $h_{ps} > 0$ or $h_{p's} > 0$. Both variables $t^s_{pp's}$ and $t^a_{pp's}$ indicate if there is tunnel between p and p' for subnet s in symmetric and asymmetric IRP respectively. In symmetric IRB, there is tunnel between p and p' if both PEs have CEs of subnet s, while in asymmetric IRB, tunnel exists even if only one of the PEs has CEs of subnet ($t^s_{pp's} \leq t^a_{pp's}$).

Going back to Table I, column # shows the number of OpenFlow entries of each group at $p \in \mathcal{P}$. There are $\sum_{s\in\mathcal{S}} h_{ps}$



(a) $\max_p f_p^s - \max_p f_p^a$ when CEs are distributed.

(b) $\max_p f_p^a - \max_p f_p^s$ when CEs are conc/ted.

(c) $(\sum_p f_p^s - \sum_p f_p^a)/P$ when CEs are distributed.

(d) $(\sum_p f_p^a - \sum_p f_p^s)/P$ when CEs are concentrated.

Fig. 3: Comparison of the number of flows entries required at highest or in average over all PEs, when either symmetric or asymmetric IRB is used.

group 1 entries, one for each local CE. The same holds for group 3 entries. Group 2 has one entry for each tunnel connecting p with another PE. Thus, in symmetric IRB, there are $\sum_{p' \in \mathcal{P}} \sum_{s \in \mathcal{S}} t^s_{pp's}$ group 2 entries for all subnets plus extra $P-1$ entries for all neutral tunnels interconnecting p with all other PEs. In asymmetric, there are $\sum_{p' \in \mathcal{P}} \sum_{s \in \mathcal{S}} t^a_{pp's}$ group 2 entries. Group 4 has $\sum_{p' \in \mathcal{P}} \sum_{s \in \mathcal{S}} t^s_{pp's} h_{p's}$ entries in symmetric IRB, since there is one entry for each remote CE belonging to the same subnet with a local CE, while in asymmetric IRB there are $\sum_{p' \in \mathcal{P}} \sum_{s \in \mathcal{S}} h_{p's}$ entries, one for each remote CE. Group 5 has as many entries as groups 3 and 4 together. Both types of IRB have one group 6 entry and one group 7 entry for each subnet, necessary for pushing the packets of each subnet to OpenFlow table 2. Group 8 exists only in symmetric IRB, since its $P-1$ entries is for the packet forwarding through the neutral tunnels to all other PEs, plus one entry for pushing to table 2 the packets coming from the other PEs. Finally, in asymmetric IRB, group 9 has one entry for each CE, while in symmetric IRB, groups 9 and 10 have one entry for each remote and local CE respectively.

Let's define f_p^s and f_p^a to be the number of OpenFlow entries existing at p in symmetric and asymmetric IRB respectively. Then

$$f_p^s = \sum_{p' \in P} \sum_{s \in S} h_{p's} + 3 \sum_{s \in S} h_{ps} + 2(P+S) - 1 + 2 \sum_{p' \in P} \sum_{s \in S} t^s_{pp's} h_{p's} + \sum_{p' \in P} \sum_{s \in S} t^s_{pp's},$$

$$f_p^a = 3 \sum_{p' \in P} \sum_{s \in S} h_{p's} + \sum_{s \in S} h_{ps} + \sum_{p' \in P} \sum_{s \in S} t^a_{pp's} + 2S.$$

Figure 3 shows a comparison between the number of flow entries required for both symmetric and asymmetric IRB, assuming that each subnet has 5 CEs. The horizontal and vertical axes give the numbers $P \in [0, 20]$ and $S \in [0, 20]$ respectively, while the color of each pixel indicates the value of the corresponding difference (white pixels correspond to negative difference). Figures 3(a)-3(b) show the difference between symmetric and asymmetric IRB in terms of the max number of flow entries over all PEs, while Figures 3(c)-3(d) show the same difference in terms of the average number of flow entries. When CEs of every subnet are distributed to as more PEs as possible, asymmetric IRB takes advantage with less number of flows either in the PE with the highest number of flow entries (Figure 3(a)) or in average over all PEs (Figure 3(c)), however, the difference is low (less than 10 flows) and happens only for low values of P or S (P should be less than the number of CEs per subnet). On the other hand, if CEs of each subnet are concentrated to a single PE, then asymmetric IRB has much more flow entries (more than 340) either in the PE with the highest number of entries (Figure 3(b)) or in average over all PEs (Figure 3(d)), where in the second case the difference is lower (almost 200).

Symmetric IRB seems to have less flow entries in the majority of CE distributions, especially when the same subnet CEs are concentrated to a single or few PEs. The existence of less number of flow entries improves the time response of the OpenFlow switch, since less entries are in a high level memory with low access speed. On the other hand, asymmetric IRB benefits from the fewer entries that packet goes through at the egress PE (2 instead of 4). In addition, it can feature less flows than symmetric IRB under some scenarios with few PEs or subnets.

V. Conclusion and Future Work

In this work, we present the benefits of each IRB solution over EVPN. Network architectures can exploit these experimentation and simulation results to define which IRB solution, either the symmetric or asymmetric one, could be better for each case. Symmetric IRB seems to offer less number of flow entries, thus smaller size of memories in the OpenFlow switches, while asymmetric IRB reduces the flow entries that each packet goes through, reducing in this way the processing time of each packet. In the future, we foresee extending our model to adapt to networks that not all PEs are equipped with IP-VRF tables.

Acknowledgment

The research leading to these results has received funding by the European Horizon 2020 Programme for research, technological development and demonstration under Grant Agreement Number 857201 (H2020 5G-VICTORI)

REFERENCES

[1] A. Sajassi, R. Aggarwal, N. Bitar, A. Isaac, J. Uttaro, J. Drake, and W. Henderickx. BGP MPLS-Based Ethernet VPN. RFC 7432, RFC Editor, February 2015. http://www.rfc-editor.org/rfc/rfc7432.txt.

[2] J. Rabadan, S. Sathappan, W. Henderickx, A. Sajassi, and J. Drake. Interconnect Solution for EVPN Overlay networks. Internet-Draft draft-ietf-bess-dci-evpn-overlay-05, IETF Secretariat, July 2017. http://www.ietf.org/internet-drafts/draft-ietf-bess-dci-evpn-overlay-05.txt.

[3] A. Sajassi, J. Drake, N. Bitar, R. Shekhar, J. Uttaro, and W. Henderickx. A Network Virtualization Overlay Solution using EVPN. Internet-Draft draft-ietf-bess-evpn-overlay-11, IETF Secretariat, January 2018. http://www.ietf.org/internet-drafts/draft-ietf-bess-evpn-overlay-11.txt.

[4] M. Mahalingam, D. Dutt, K. Duda, P. Agarwal, L. Kreeger, T. Sridhar, M. Bursell, and C. Wright. Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. RFC 7348, RFC Editor, August 2014. http://www.rfc-editor.org/rfc/rfc7348.txt.

[5] A. Sajassi, S. Salam, S. Thoria, J. Drake, and J. Rabadan. Integrated Routing and Bridging in EVPN. Internet-Draft draft-ietf-bess-evpn-inter-subnet-forwarding-05, IETF Secretariat, July 2018. http://www.ietf.org/internet-drafts/draft-ietf-bess-evpn-inter-subnet-forwarding-05.txt.

[6] Ryu. https://osrg.github.io/ryu/.

[7] Network Implementation Testbed using Open Source platforms. https://nitlab.inf.uth.gr/NITlab/nitos.

[8] Open vSwitch. https://www.openvswitch.org/.

[9] Mininet. http://mininet.org/.