

**BGP accept-own-next-hop community attribute  
draft-agrewal-idr-accept-own-next-hop-00**

Abstract

Various Service chain techniques utilize a Controller to inject Border Gateway Protocol (BGP) Virtual Private Network (VPN) routes to help steer traffic through a given path. The Controller does so by controlling how these VPN routes are imported into various Virtual Routing and Forwarding (VRF) tables at routers along the desired path. A couple of such approaches are specified in [[I-D.ietf-bess-service-chaining](#)]. These approaches rely on the Controller modifying the Route Target (RT) list and next-hop of a VPN route received from a downstream router and redistributing these modified routes to upstream routers. This is done such that -

- o routes originated by an ingress VRF at the downstream router are imported into the egress VRF at the immediately preceding upstream router and
- o next-hop advertised to the upstream router is the address of the immediately succeeding downstream router.

This forces the traffic to flow through a sequence of network functions creating a service chain.

This works fine as long as the VRF importing the route received from the Controller is on a different router than the VRF that originally exported the route to the Controller. This is because BGP protocol [[RFC4271](#)] specifies that a router reject routes received with its own next-hop. This document proposes a new community the reception of which relaxes this particular rule in the BGP protocol standard and describes at least one way of how next-hops of such routes could be resolved.

Requirements Language

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

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

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

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

This Internet-Draft will expire on April 22, 2018.

Copyright Notice

Copyright (c) 2017 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 (<https://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.](#) Use Case . . . . . [3](#)
- [3.](#) accept-own-nexthop community . . . . . [5](#)
  - [3.1.](#) New BGP behavior . . . . . [5](#)
  - [3.2.](#) Configuration Control . . . . . [5](#)
- [4.](#) IANA Considerations . . . . . [5](#)
- [5.](#) Security Considerations . . . . . [5](#)
- [6.](#) Acknowledgements . . . . . [6](#)
- [7.](#) References . . . . . [6](#)
  - [7.1.](#) Normative References . . . . . [6](#)
  - [7.2.](#) Informative References . . . . . [6](#)
- Authors' Addresses . . . . . [7](#)



## **1. Introduction**

The BGP ACCEPT\_OWN Community Attribute standard [[RFC7611](#)] defined a technique to let the route reflector modify the RT list of a VPN route and redistribute it back to the originating Provider Edge (PE). This enables the route reflector to control how a route originated within one VRF at the PE is imported into other VRFs at the originating PE. However, the ACCEPT\_OWN standard does not specify how the forwarding next-hop for such an accepted route is derived. It is understood that once the source VRF is located, the original route for this prefix in this source VRF is located and the next-hop information of the original route is used as the forwarding state for the received route. The standard also mandates that the route received from the route reflector be originated by a source VRF on the receiving router.

This document proposes a new community - BGP accept-own-nexthop - whose usage means - 'accept a route with one's own next-hop' (regardless of whether the route was originated by the PE). Doing so, enables some new useful use cases. The scope of this community does not mandate the route be originated by a source VRF on the receiving router as was the case in BGP ACCEPT\_OWN community attribute. Also, the usage of this community does not specify how the forwarding next-hop of such a route is derived. One different approach, than the implicit approach used in BGP ACCEPT\_OWN standard, to obtain the forwarding information of the received route is described below.

## **2. Use Case**

Let us walk through an example of how a router could redirect traffic, that it is receiving in the Ingress VRF, towards a Layer 2 Physical Network Function (PNF) before the traffic exits this router. A gateway in the form of a router is necessary to have Layer 2 PNFs become part of a service chain.



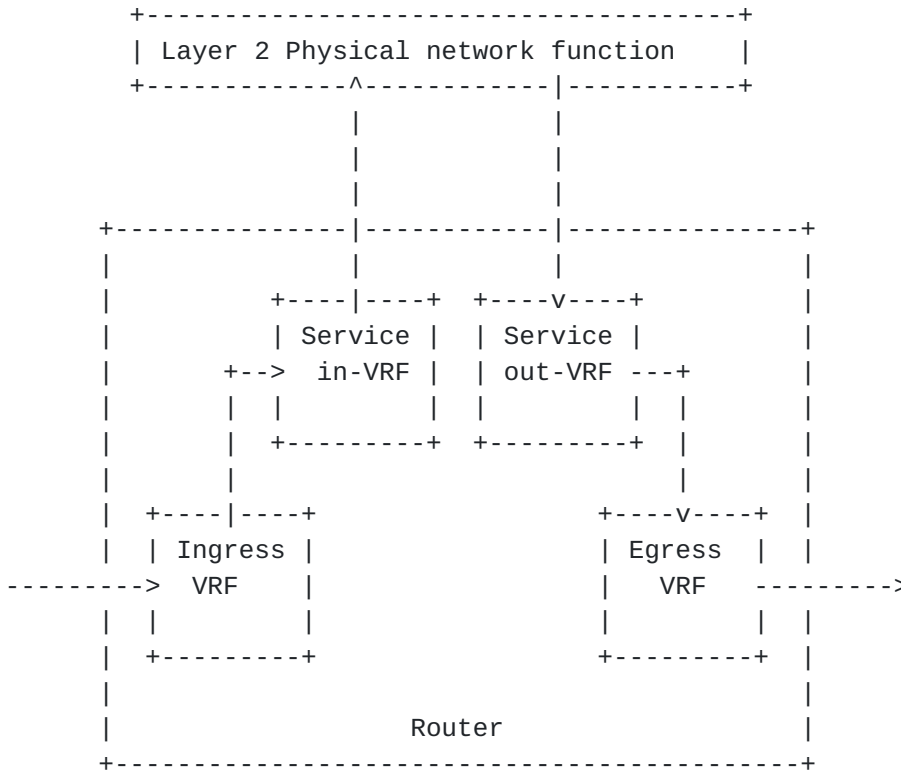


Figure 1: A Layer 2 PNF as part of service chain

In the service chaining scenario involving a Layer 2 PNF - a PE (Fig. 1) advertises a static route - configured inside a VRF (service in-VRF) - whose next-hop is the interface that a (transparent layer-2) service instance (viz., firewall, anti-virus system etc.) is connected to. The prefix for this route is an address that also belongs to the same subnet as this interface's address. The destination address will be in a different vrf (Service out-VRF) at the PE itself where the serviced traffic coming from the service will be received to be sent to it's destination.

PE will advertise this route with next-hop as itself and a label that identifies the interface on the PE that the service is connected to.

A Controller acting as an extended route reflector could now steer the traffic for a particular destination - for which the controller wants the traffic to be serviced - by sending a VPN route for that destination, with PE's own address as the next-hop and the above label. The RT that Controller attaches to such a route would allow it to be imported to the Ingress VRF. Such a route will be resolved using the received VPN label - that the PE itself advertised - to point to whatever next-hop information that is associated with this label locally. A controller may create multiple such service VRFs on the PE and follow the above procedure for each service to construct a



service chain by advertising routes with different RTs for the same destination prefix with different labels, where each label identifies the interface towards that particular service.

### **3. accept-own-nexthop community**

A new well-known BGP community in the First Come First Served [[RFC5226](#)] range called accept-own-nexthop has been assigned value of 0XFFF008 by IANA.

#### **3.1. New BGP behavior**

A change to default BGP behavior is proposed such that a router that receives a route, whose NEXT\_HOP value matches one of the addresses configured on itself, MAY accept the route if and only if the following are true:

- o The received route is carrying the accept-own-nexthop community.
- o Processing of the accept-own-nexthop community is enabled by configuration on the receiving router.

#### **3.2. Configuration Control**

The processing - as defined above - of accept-own-nexthop community is disabled by default. An implementation SHOULD provide a configuration statement to enable a router to activate the behavior specified in this document.

### **4. IANA Considerations**

IANA has assigned the value 0xFFFF0008 in the "BGP Well-known Communities" registry for the accept-own-nexthop community.

### **5. Security Considerations**

accept-own-nexthop community allows a router to accept a route with it's own next-hop. If the originator of that route is that router itself and if the router accepts the received route to the same VRF from where it was originated route oscillations would happen if this new route is more preferable than the original route. That is so because the receiving router preferring the received route would lead to it withdrawing its advertisement for the original route. This will prompt the Controller to withdraw the re-originated route. This in turn will prompt the PE to re-advertise the original route and the cycle would continue.





Since these routes are like any other BGP VPN route, all the vulnerabilities applicable to any other BGP VPN route are also applicable to these routes. Such vulnerabilities for BGP VPN routes have been described in [[RFC4364](#)].

## **6. Acknowledgements**

The authors would like to thank John Scudder, Jeff Haas and Minto Jeyanath for their valuable comments and suggestions.

## **7. References**

### **7.1. Normative References**

- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", [RFC 4271](#), DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", [RFC 4364](#), DOI 10.17487/RFC4364, February 2006, <<https://www.rfc-editor.org/info/rfc4364>>.

### **7.2. Informative References**

- [I-D.ietf-bess-service-chaining] Fernando, R., Mackie, S., Rao, D., Rijsman, B., Napierala, M., and T. Morin, "Service Chaining using Virtual Networks with BGP VPNs", [draft-ietf-bess-service-chaining-03](#) (work in progress), July 2017.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [RFC 5226](#), DOI 10.17487/RFC5226, May 2008, <<https://www.rfc-editor.org/info/rfc5226>>.
- [RFC7611] Uttaro, J., Mohapatra, P., Smith, D., Raszuk, R., and J. Scudder, "BGP ACCEPT\_OWN Community Attribute", [RFC 7611](#), DOI 10.17487/RFC7611, August 2015, <<https://www.rfc-editor.org/info/rfc7611>>.



Authors' Addresses

Ashutosh Grewal  
Juniper Networks  
1133 Innovation Way  
Sunnyvale, CA 94089  
USA

E-Mail: [agrewal@juniper.net](mailto:agrewal@juniper.net)

Nischal Sheth  
Juniper Networks  
1133 Innovation Way  
Sunnyvale, CA 94089  
USA

E-Mail: [nsheth@juniper.net](mailto:nsheth@juniper.net)

Kaliraj Vairavakkalai  
Juniper Networks  
1133 Innovation Way  
Sunnyvale, CA 94089  
USA

E-Mail: [kaliraj@juniper.net](mailto:kaliraj@juniper.net)

