NVO3 Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 24, 2020

X. Min
G. Mirsky
ZTE Corp.
S. Pallagatti
VMware

October 22, 2019

BFD for Geneve
draft-xiao-nvo3-bfd-geneve-01

Abstract

This document describes the use of the Bidirectional Forwarding Detection (BFD) protocol in point-to-point Generic Network Virtualization Encapsulation (Geneve) tunnels forming up an overlay network.

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 24, 2020.

Table of Contents

1. Introduction

"Generic Network Virtualization Encapsulation" (Geneve) [I-D.ietf-nvo3-geneve] provides an encapsulation scheme that allows building an overlay network by decoupling the address space of the attached virtual hosts from that of the network.

This document describes the use of Bidirectional Forwarding Detection (BFD) protocol [RFC5880] to enable monitoring continuity of the path between two Geneve tunnel endpoints, which may be NVE (Network Virtualization Edge) or other device acting as a Geneve tunnel endpoint. For simplicity, in this document, NVE is used to represent Geneve tunnel endpoint, TS (Tenant System) is used to represent the physical or virtual device attached to a Geneve tunnel endpoint from the outside. VAP (Virtual Access Point) is the NVE side of the interface between the NVE and the TS, and a VAP is a logical network port (virtual or physical) into a specific virtual network. For detailed definitions and descriptions of NVE, TS and VAP, please refer to [RFC7365] and [RFC8014].

The use cases and the deployment of BFD for Geneve are consistent with what's described in Section 1 and Section 3 of

[I-D.ietf-bfd-vxlan]. The major difference between Geneve and "Virtual eXtensible Local Area Network" (VXLAN) [RFC7348] encapsulation is that Geneve supports multi-protocol payload and variable length options.

2. Conventions Used in This Document

2.1. Terminology

BFD: Bidirectional Forwarding Detection

CC: Continuity Check

GAL: Generic Associated Channel Label

G-ACh: Generic Associated Channel

Geneve: Generic Network Virtualization Encapsulation

MPLS: Multiprotocol Label Switching

NVE: Network Virtualization Edge

TS: Tenant System

VAP: Virtual Access Point

VNI: Virtual Network Identifier

VXLAN: Virtual eXtensible Local Area Network

2.2. Requirements Language

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

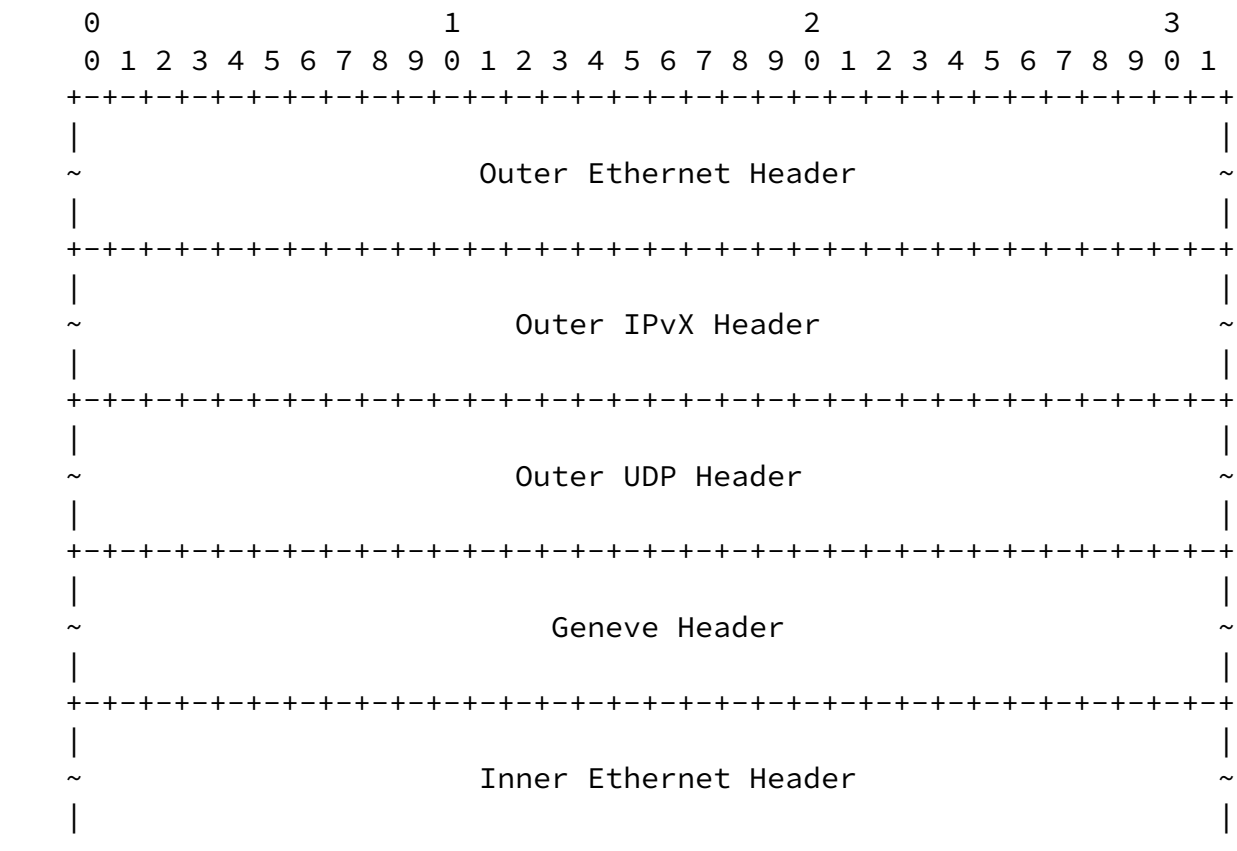
3. BFD Packet Transmission over Geneve Tunnel

Concerning whether or not the Geneve data packets include an IP protocol data unit, and whether or not the Geneve data packets include an MPLS protocol data unit, this document considers three options of BFD packet encapsulation in Geneve.

3.1. BFD Encapsulation With Inner Ethernet/IP/UDP Header

If the Protocol Type field (as defined in Section 3.4 of [I-D.ietf-nvo3-geneve]) of data packets indicates that there exists an inner Ethernet header, i.e., the Protocol Type equals to 0x6558 (Ethernet frame), then BFD packets are encapsulated in Geneve as described below. The Geneve packet format over IPv4 is defined in

Section 3.1 of [I-D.ietf-nvo3-geneve]. The Geneve packet format over IPv6 is defined in Section 3.2 of [I-D.ietf-nvo3-geneve]. The Outer IP/UDP and Geneve headers MUST be encoded by the sender as defined in [I-D.ietf-nvo3-geneve]. Note that the outer IP header and the inner IP header may not be of the same address family, in other words, outer IPv6 header accompanied with inner IPv4 header and outer IPv4 header accompanied with inner IPv6 header are both possible.


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
Outer IPvX Header
Outer UDP Header
Geneve Header
Inner Ethernet Header

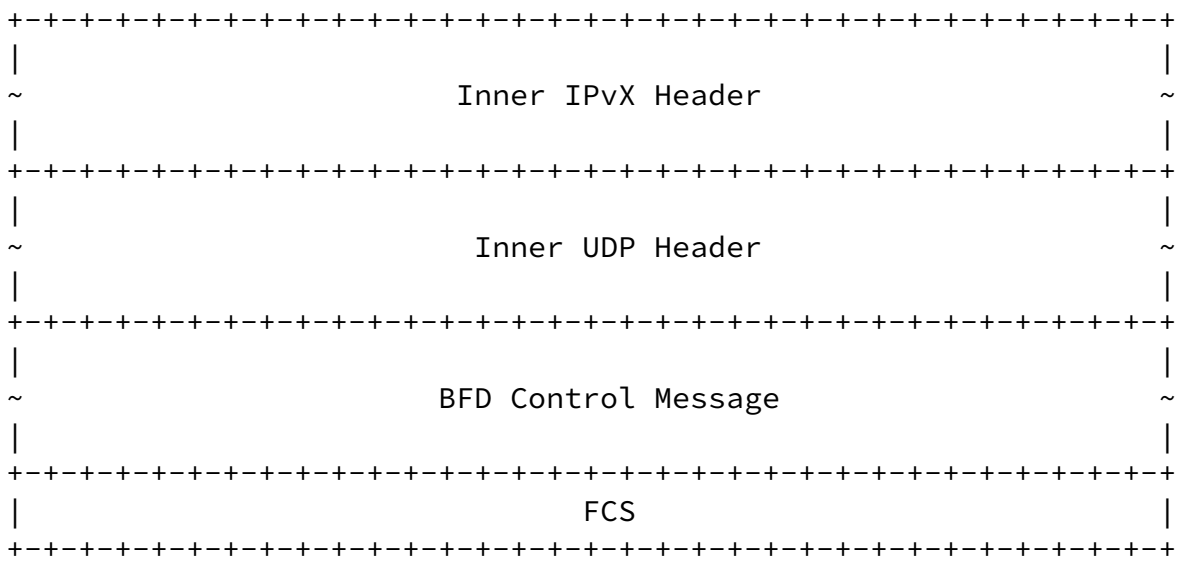


Figure 1: Geneve Encapsulation of BFD Control Message With the Inner Ethernet/IP/UDP Header

The BFD packet MUST be carried inside the inner Ethernet frame of the Geneve packet, as specified in Section 4 of [I-D.ietf-bfd-vxlan].

When the BFD packets are encapsulated in Geneve in this way, the Geneve header follows the value set below.

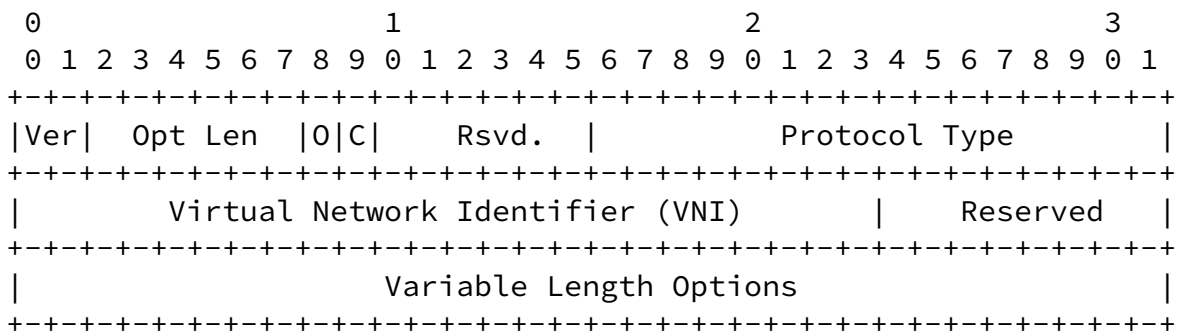


Figure 2: Geneve Header

Opt Len field SHOULD be set to 0, which indicates there isn't any variable length option.

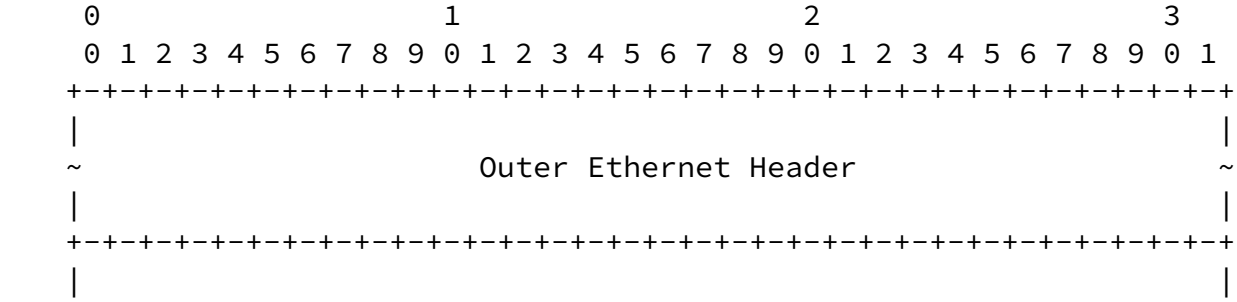
O bit MUST be set to 1, which indicates this packet contains a control message.

C bit MUST be set to 0.

Protocol Type field MUST be set to 0x6558 (Ethernet frame).

3.2. BFD Encapsulation With Inner IP/UDP Header

If the Protocol Type field (as defined in Section 3.4 of [I-D.ietf-nvo3-geneve]) of data packets indicates that there exists an inner IP header, i.e., the Protocol Type equals to 0x0800 (IPv4) or 0x86DD (IPv6), then BFD packets are encapsulated in Geneve as described below. The Geneve packet format over IPv4 is defined in Section 3.1 of [I-D.ietf-nvo3-geneve]. The Geneve packet format over IPv6 is defined in Section 3.2 of [I-D.ietf-nvo3-geneve]. The Outer IP/UDP and Geneve headers MUST be encoded by the sender as defined in [I-D.ietf-nvo3-geneve]. Note that the outer IP header and the inner IP header may not be of the same address family, in other words, outer IPv6 header accompanied with inner IPv4 header and outer IPv4 header accompanied with inner IPv6 header are both possible.



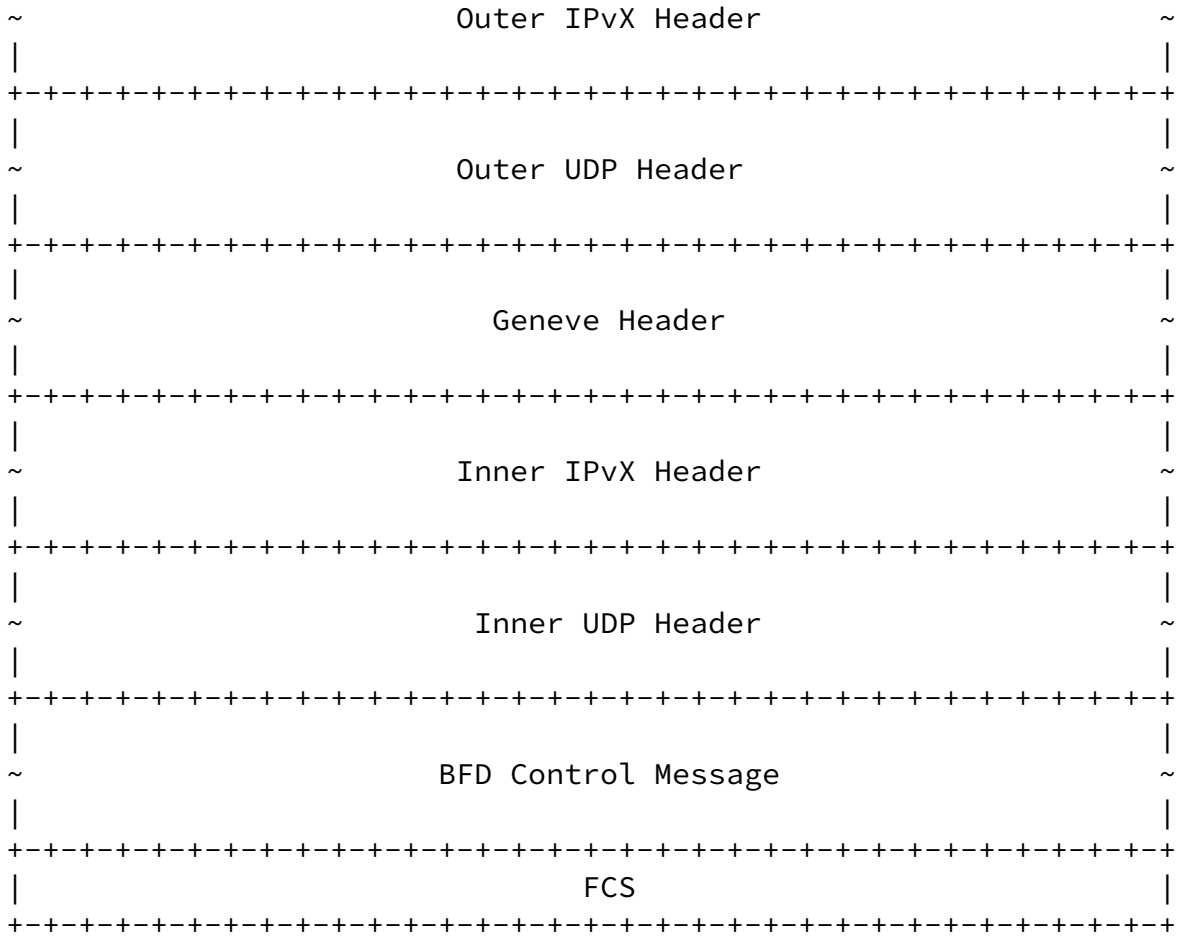


Figure 3: Geneve Encapsulation of BFD Control Message With the Inner IP/UDP Header

The BFD packet MUST be carried inside the inner IP packet of the Geneve packet. The inner IP packet carrying the BFD payload has the following format:

IP header:

Source IP: IP address of a VAP of the originating NVE.

Destination IP: IP address of a VAP of the terminating NVE.

TTL: MUST be set to 1 to ensure that the BFD packet is not

routed within the L3 underlay network.

The fields of the UDP header and the BFD control packet are encoded as specified in [RFC5881].

When the BFD packets are encapsulated in Geneve in this way, the Geneve header follows the value set below.

Opt Len field SHOULD be set to 0, which indicates there isn't any variable length option.

O bit MUST be set to 1, which indicates this packet contains a control message.

C bit MUST be set to 0.

Protocol Type field MUST be set to 0x0800 (IPv4) or 0x86DD (IPv6), depending on the address family of the inner IP packet.

3.3. BFD Encapsulation With Inner MPLS Header

If the Protocol Type field (as defined in Section 3.4 of [I-D.ietf-nvo3-geneve]) of data packets indicates that there exists an inner MPLS header, i.e., the Protocol Type equals to 0x8847 (MPLS) or 0x8848 (MPLS with the upstream-assigned label), then BFD packets are encapsulated in Geneve as described below. The Geneve packet format over IPv4 is defined in Section 3.1 of [I-D.ietf-nvo3-geneve]. The Geneve packet format over IPv6 is defined in Section 3.2 of [I-D.ietf-nvo3-geneve]. The Outer IP/UDP and Geneve headers MUST be encoded by the sender as defined in [I-D.ietf-nvo3-geneve].

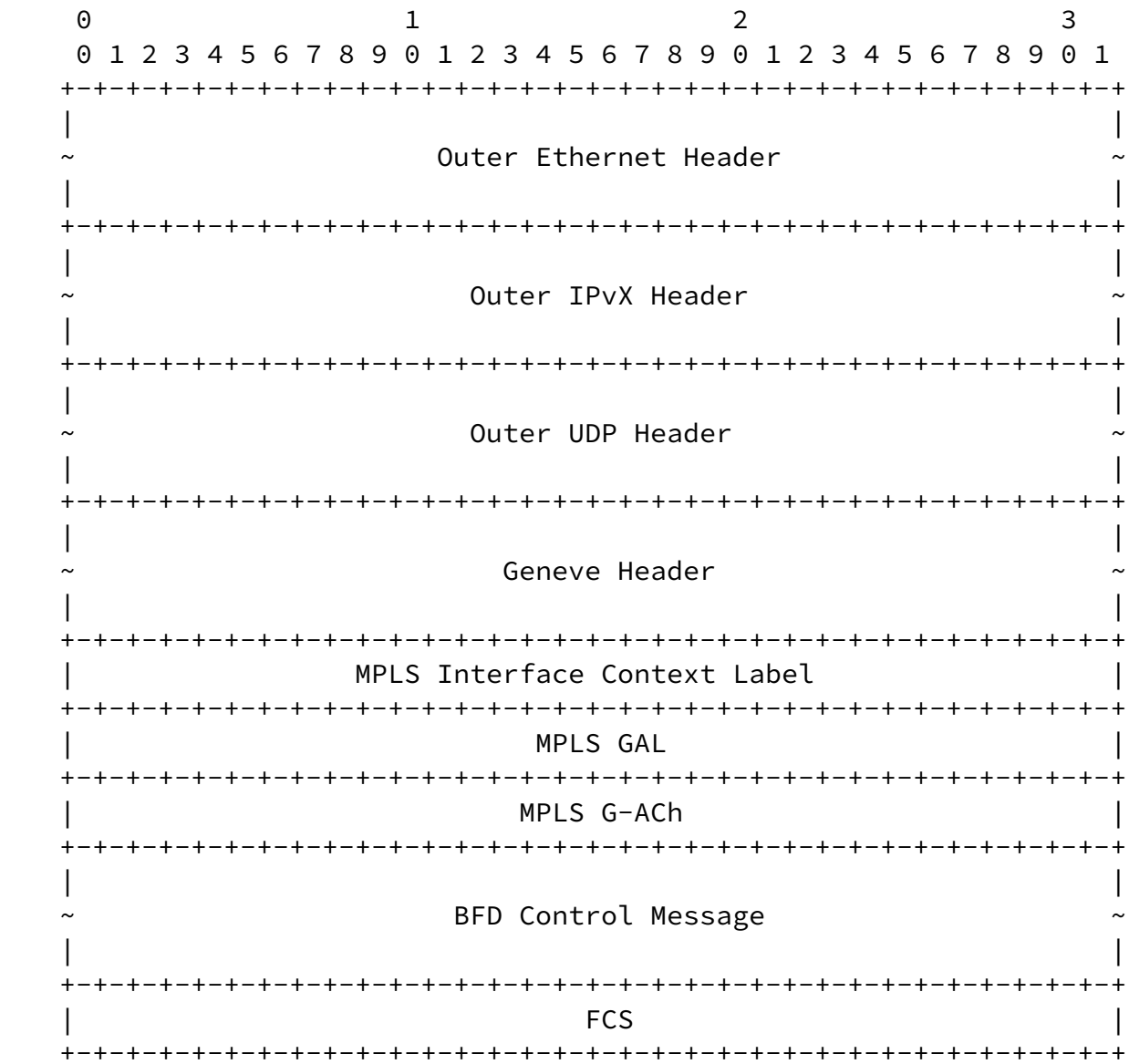


Figure 4: Geneve Encapsulation of BFD Control Message With the Inner MPLS GAL/G-ACh

The BFD packet MUST be carried inside the inner MPLS packet of the Geneve packet. The inner MPLS packet carrying the BFD payload has the following format:

MPLS Interface Context Label: This Label would be used to identify a VAP of the originating NVE and a VAP of the terminating NVE.

MPLS GAL (Generic Associated Channel Label):

Label value: MUST be set to 13, as specified in [RFC5586].

S bit: MUST be set to 1.

TTL: MUST be set to 1.

The fields of the MPLS G-ACh (Generic Associated Channel) and the BFD control packet are encoded as specified for MPLS-TP CC (Continuity Check) message in [RFC6428].

When the BFD packets are encapsulated in Geneve in this way, the Geneve header follows the value set below.

Opt Len field SHOULD be set to 0, which indicates there isn't any variable length option.

O bit MUST be set to 1, which indicates this packet contains a control message.

C bit MUST be set to 0.

Protocol Type field MUST be set to 0x8847 (MPLS).

4. Reception of BFD packet from Geneve Tunnel

Once a packet is received, NVE MUST validate the packet as described in [I-D.ietf-nvo3-geneve].

If the Protocol Type field equals 0x6558 (Ethernet frame), and the Destination MAC of the inner Ethernet frame matches one MAC address owned by the NVE, the Destination IP, the UDP destination port and the TTL of the inner IP packet MUST be validated to determine whether the received packet can be processed by BFD. BFD packet with inner MAC set to NVE MUST NOT be forwarded to TSs.

If the Protocol Type field equals 0x0800 (IPv4) or 0x86DD (IPv6), and the Destination IP of the inner IP packet matches a VAP IP address of the NVE, the UDP destination port and the TTL of the inner IP packet MUST be validated to determine whether the received packet can be processed by BFD. BFD packet with inner IP set to NVE MUST NOT be forwarded to TSs.

If the Protocol Type field equals 0x8847 (MPLS), the MPLS

Interface Context Label, the MPLS GAL and the MPLS G-ACh of the inner MPLS packet MUST be validated to determine whether the received packet can be processed by BFD. BFD packet with MPLS GAL MUST NOT be forwarded to TSs.

4.1. Demultiplexing of the BFD packet

In BFD over Geneve, a BFD session is originated and terminated at VAP, usually one NVE owns multiple VAPs, so multiple BFD sessions may be running between two NVEs, there needs to be a mechanism for demultiplexing received BFD packets to the proper session.

If the BFD packet is received with Your Discriminator equals to 0, for different BFD encapsulation, the procedure for demultiplexing the received BFD packets is different.

When the BFD Encapsulation With Inner Ethernet/IP/UDP Header is used, the BFD session MUST be identified using the procedure specified in Section 5.1 of [I-D.ietf-bfd-vxlan].

When the BFD Encapsulation With Inner IP/UDP Header is used, the BFD session MUST be identified using the inner IP/UDP header, i.e., the source IP and the destination IP present in the inner IP/UDP header.

When the BFD Encapsulation With Inner MPLS Header is used, the BFD session MUST be identified using the inner MPLS header, i.e., the MPLS Interface Context Label present in the inner MPLS header.

If the BFD packet is received with non-zero Your Discriminator, then BFD session MUST be demultiplexed only with Your Discriminator as the key.

With respect to BFD for Geneve, the use of the specific VNI would follow the principle as specified in Section 6 of [I-D.ietf-bfd-vxlan].

[Ed.Note]: Currently it's still undetermined whether "BFD for VxLAN" should allow multiple BFD sessions for the same VNI. The Editor leans to believe "BFD for Geneve" should allow multiple BFD sessions for the same VNI, and it needs further discussion.

5. Security Considerations

This document does not raise any additional security issues beyond those of the specifications referred to in the list of normative references.

6. IANA Considerations

This document has no IANA action requested.

7. Acknowledgements

The authors would like to acknowledge Reshad Rahman, Jeffrey Haas and Matthew Bocci for their guidance on this work.

8. Normative References

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

[I-D.ietf-nvo3-geneve]
Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-14 (work in progress), September 2019.

[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>.

[RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009,

<https://www.rfc-editor.org/info/rfc5586>.

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

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

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

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7365] Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for Data Center (DC) Network Virtualization", RFC 7365, DOI 10.17487/RFC7365, October 2014, <https://www.rfc-editor.org/info/rfc7365>.

[RFC8014] Black, D., Hudson, J., Kreeger, L., Lasserre, M., and T. Narten, "An Architecture for Data-Center Network Virtualization over Layer 3 (NVO3)", RFC 8014, DOI 10.17487/RFC8014, December 2016, <https://www.rfc-editor.org/info/rfc8014>.

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

Authors' Addresses

Xiao Min

ZTE Corp.
Nanjing
China

Phone: +86 25 88013062
Email: xiao.min2@zte.com.cn

Greg Mirsky
ZTE Corp.
USA

Email: gregimirsky@gmail.com

Santosh Pallagatti
VMware

Email: santosh.pallagatti@gmail.com