NVO3 Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 19, 2018

G. Mirsky
ZTE Corp.
N. Kumar
D. Kumar
Cisco Systems, Inc.
M. Chen
Y. Li
Huawei Technologies
D. Dolson
Sandvine
March 18, 2018

OAM Header for use in Overlay Networks
draft-ooamdt-rtgwg-ooam-header-04

Abstract

This document introduces Overlay Operations, Administration, and Maintenance (OOAM) Header to be used in overlay networks to create Overlay Associated Channel (OAC) to ensure that OOAM control packets are in-band with user traffic and de-multiplex OOAM protocols.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

New protocols that support overlay networks like VxLAN-GPE [I-D.ietf-nvo3-vxlan-gpe], GUE [I-D.ietf-nvo3-gue], Geneve [I-D.ietf-nvo3-geneve], BIER [RFC8296], and NSH [RFC8300] support multi-protocol payload, e.g. Ethernet, IPv4/IPv6, and recognize Operations, Administration, and Maintenance (OAM) as one of distinct types. That ensures that Overlay OAM (OOAM)packets are sharing fate with Overlay data packet traversing the underlay.

This document introduces generic requirements to OAM protocols used in overlay networks and defines OOAM Header to be used in overlay networks to de-multiplex OOAM protocols.

1.1. Conventions used in this document

1.1.1. Terminology

Term "Overlay OAM" used in this document interchangeably with longer version "set of OAM protocols, methods and tools for Overlay networks".

NTP Network Time Protocol

OAC Overlay Associated Channel

OAM Operations, Administration, and Maintenance

OOAM Overlay OAM

PTP Precision Time Protocol

1.1.2. 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 [RFC2119].

2. General Requirements to OAM Protocols in Overlay Networks

OAM protocols, whether it is part of fault management or performance monitoring, intended to provide reliable information that can be used to identify defect, localize it and apply corrective actions. One of the main challenges that network operators may encounter is interpretations of reports of the defect or service degradation and correlation to affected services. In order to improve reliability of the correlation process we set forth the following requirements:

REQ#1: Overlay OAM packets SHOULD be fate sharing with data traffic, i.e. in-band with the monitored traffic, i.e. follow exactly the same overlay and transport path as data plane traffic, in forward direction, i.e. from ingress toward egress end point(s) of the OAM test.

REQ#2: Encapsulation of OAM control message and data packets in underlay network MUST be indistinguishable from underlay network forwarding point of view.

REQ#3: Presence of OAM control message in overlay packet MUST be unambiguously identifiable.

REQ#4: It MUST be possible to express entropy for underlay Equal Cost Multipath in overlay encapsulation in order to avoid using data packet content by underlay transient nodes.

3. Associated Channel in Overlay Networks

Associated channel in the overlay network is the channel that, by using the same encapsulation as user traffic, follows the same path through the underlay network as user traffic. In other words, the associated channel is in-band with user traffic. Creating notion of the overlay associated channel (OAC) in the overlay network ensures that control packets of active OAM protocols carried in the OAC are in-band with user traffic. Additionally, OAC allows development of OAM tools that, from operational point of view, function in essentially the same manner in any type of overlay.

4. Overlay OAM Header

OOAM Header immediately follows the header of the overlay and identifies OAC. The format of the OOAM Header is:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| V |           Msg Type        |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Flags             |    Reserved   |   Next Prot   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   OOAM control message                        ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Overlay OAM Header format

The OAM Header consists of the following fields:

- V - two bits long field indicates the current version of the Overlay OAM Header. The current value is 0;

- Msg Type - 14 bits long field identifies OAM protocol, e.g. Echo Request/Reply, BFD, Performance Measurement;

- Length - two octets long field that is length of the OOAM control packet in octets;

- Flags -two octets long field carries bit flags that define optional capability and thus processing of the OOAM control packet;

- Reserved - one octet field that MUST be zeroed on transmit and ignored on receipt;

- Next Prot - one octet long field that defines optional payload that is present after the OOAM Control Packet.

The format of the Flags field is:

```
  0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|T|          Reserved           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Flags field format

where:

- T - Timestap block flag.

- Reserved - must be set to all zeroes on transmission and ignored on receipt.

The OOAM header may be followed by the Timestamp control block Figure 3 and then by OOAM Control Packet identified by the Msg Type field.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  QTF  |  RTF  |                   Reserved                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Timestamp 1                         |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Timestamp 4                         |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Timestamp block format

where:

QTF - Querier timestamp format

RTF - Responder timestamp format

Timestamp 1-4 - 64-bit timestamp values

Network Time Protocol (NTP), described in [RFC5905], is widely used and has long history of deployment. But it is the IEEE 1588 Precision Time Protocol (PTP) [IEEE.1588.2008] that is being broadly used to achieve high-quality clock synchronization. Converging between NTP and PTP time formats is possible but is not trivial and does come with cost, particularly when it is required to be performed in real time without loss of accuracy. And recently protocols that supported only NTP time format, like One-Way Active Measurement Protocol [RFC4656] and Two-Way Active Measurement Protocol [RFC5357], have been enhanced to support the PTP time format as well [RFC8186]. This document proposes to select PTP time format as default time format for Overlay OAM performance measurement. Hence QTF, RTF fields MUST be set to 0 if querier or responder use PTP time format respectively. If the querier or responder use the NTP time format, then QTF and/or RTF MUST be set to 1. Use of other values MUST be considered as error and MAY be reported.

4.1. Use of OOAM Header in Active OAM

Active OAM methods, whether used for fault management or performance monitoring, generate dedicated test packets [RFC7799]. Format of an OAM test packet in overlay network presented in Figure 4.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~              Underlay network encapsulation                   ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~              Overlay network encapsulation                    ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                     OOAM Header                +-+-+-+-+-+-+-+-+
   |                                                |NextProt = None|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                  OOAM control message                         ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Overlay OAM Header in Active OAM Control Packet

Because active OAM method uses only OAM protocol value of Next Prot field in the OOAM header is set to None indicating that there’s no content from other protocol immediately after OOAM control message in the packet.

4.2. Use of OOAM Header in Hybrid OAM

Hybrid OAM Type I methods, whether used for fault management or performance monitoring, modify user data packets [RFC7799]. Format of such modified packet in overlay network presented in Figure 5.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~               Underlay network encapsulation                  ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~               Overlay network encapsulation                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                      OOAM Header             +-+-+-+-+-+-+-+-+
|                                              |NextProt = Data|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   OOAM control message                        ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                          User data                            ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: Overlay OAM Header in Hybrid OAM Control Packet

In case when OOAM header used for Hybrid Type I OAM method value of the Next Prot field is set to the value associated with the protocol of the user data.

5. IANA Considerations

IANA is requested to create new registry called "Overlay OAM".

5.1. OOAM Message Types

IANA is requested to create new sub-registry called "Overlay OAM Protocol Types" in the "Overlay OAM" registry. All code points in the range 1 through 15615 in this registry shall be allocated according to the "IETF Review" procedure as specified in [RFC8126] . Remaining code points are allocated according to the Table 1:

Value	Description	Reference
0	Reserved	
1 - 15615	Unassigned	IETF Review
15616 - 16127	Unassigned	First Come First Served
16128 - 16143	Experimental	This document
16144 - 16382	Private Use	This document
16383	Reserved	This document

Table 1: Overlay OAM Protocol type

5.2. OOAM Header Flags

IANA is requested to create sub-registry "Overlay OAM Header Flags" in "Overlay OAM" registry. Two flags are defined in this document. New values are assigned via Standards Action [RFC8126].

Flags bit	Description	Reference
Bit 0	Timestamp field	This document
Bit 1-15	Unassigned	

Table 2: Overlay OAM Flags

6. Security Considerations

TBD

7. Contributors

Work on this documented started by Overlay OAM Design Team with contributions from:

Carlos Pignataro

Cisco Systems, Inc.

cpignata@cisco.com

Erik Nordmark

Arista Networks

nordmark@acm.org

Ignas Bagdonas

ibagdona@gmail.com

David Mozes

Mellanox Technologies Ltd.

davidm@mellanox.com

8. Acknowledgement

TBD

9. References

9.1. Normative References

[IEEE.1588.2008]
"Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE Standard 1588, July 2008.

[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>.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <https://www.rfc-editor.org/info/rfc5905>.

9.2. Informative References

[I-D.ietf-nvo3-geneve]
Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-06 (work in progress), March 2018.

[I-D.ietf-nvo3-gue]
Herbert, T., Yong, L., and O. Zia, "Generic UDP Encapsulation", draft-ietf-nvo3-gue-05 (work in progress), October 2016.

[I-D.ietf-nvo3-vxlan-gpe]
Maino, F., Kreeger, L., and U. Elzur, "Generic Protocol Extension for VXLAN", draft-ietf-nvo3-vxlan-gpe-05 (work in progress), October 2017.

[RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <https://www.rfc-editor.org/info/rfc4656>.

[RFC5357] Hedayat, K., Krzanowski, R., Morton, A., Yum, K., and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)", RFC 5357, DOI 10.17487/RFC5357, October 2008, <https://www.rfc-editor.org/info/rfc5357>.

[RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <https://www.rfc-editor.org/info/rfc7799>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8186] Mirsky, G. and I. Meilik, "Support of the IEEE 1588 Timestamp Format in a Two-Way Active Measurement Protocol (TWAMP)", RFC 8186, DOI 10.17487/RFC8186, June 2017, <https://www.rfc-editor.org/info/rfc8186>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

Authors’ Addresses

Greg Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Nagendra Kumar
Cisco Systems, Inc.

Email: naikumar@cisco.com

Deepak Kumar
Cisco Systems, Inc.

Email: dekumar@cisco.com

Mach Chen
Huawei Technologies

Email: mach.chen@huawei.com

Yizhou Li
Huawei Technologies

Email: liyizhou@huawei.com

David Dolson
Sandvine

Email: ddolson@sandvine.com