NVO3 WG
Internet-Draft
Intended status: Standards Track
Expires: August 31, 2018

T. Ao
ZTE Corporation
G. Mirsky
ZTE Corp.
Y. Fan
China Telecom

February 27, 2018

# Multi-encapsulation interconnection for Overlay Virtual Network
draft-ao-nvo3-multi-encap-interconnect-00

## Abstract

For an virtualized overlay network, there are many encapsulations that may be used. Different customer have their own preference. So if some of these different encapsulation can be interconnected together, the virtualized overlay network would be more compatible and have loose strict on access. This document is going to provide an architecture of different overlay encapsulation interconnection and an tranformer gateway for these end station connected to the virtual 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 August 31, 2018.

## Copyright Notice

Table of Contents



## 1. Introduction

Network virtualization using Overlays over Layer 3 (NVO3) is a technology that is used to address issues that arise in building large, multi-tenant data centers that make extensive use of server virtualization.

With the progress in NVO3 WG, some of the data plane encapsulations have been put forward, some are outstanding dataplane for overlay network, such as VxLAN-GPE [I-D.ietf-nvo3-vxlan-gpe], GENEVE [I-D.ietf-nvo3-geneve] and GUE [I-D.ietf-nvo3-encap], etc. The consideration about these overlay encapsulations has been analysised in [I-D.ietf-nvo3-encap]. The fact is that each of them have its custmers, and furthermore, some of them have been provisioned in the network. So that a problem arises: for a virtual network, all the hosts that connect to the same VN and want to communicate with each other are required to have the same data plane encapsulation. This problem limite the network scalability and capacity. Especially, when the NVE is located on the vSwitch, the encapsulation method on the NVE is not predictable. Allowing as many kinds of accession as possible is more attractive for a virtualized overlay network.

To improve the scalability and capacity of the virtualized overlay network, we propose a multi-encapsulation access allowed interconnect NVO3 architecture, and a gateway in the network to perform the transformation for different encapsulation in this document, by which these hosts with different encapsulations can be interconnected. Here we call the gateway as Transformer Gateway in the following descrption.

## 2. Conventions used in this document

### 2.1. Terminology

NVO3: Network Virtualization using Overlay over Layer 3

NVA: Network Virtualization Authority

TS: Tenant System

VxLAN-GPE: Virtual extension LAN with Generic Protocol Extension

GENEVE: Generic Network Virtualization Encapsulation

GUE: Generic UDP Encapsulation

Multi-encapsulation NVO: an virtualized overlay network that allow multiple different encapsulations interconnection.

t-GW: Tranformation Gateway. A gateway that do the transformer for different encapsulation to make them can communicate with each other.

tNVE: A NVE that complete the functionn of a tranformer gateway.

### 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. Multi-encapsulation NVO architecture

In the multi-encapsulation interconnection allowed NVO, different NVE may support different encapsulation data plane. As we have know that any two of the TS in the same VN should communicate with each other, but it is required that both of overaly encapsulation they are using to access to the virtual network have to be same. In order to relieve the limitation and to support these encapsulation would

interconnect together, a multi-encapsulation architecture is introdueced. Figure 1 dipcts a reference architecture in VN.

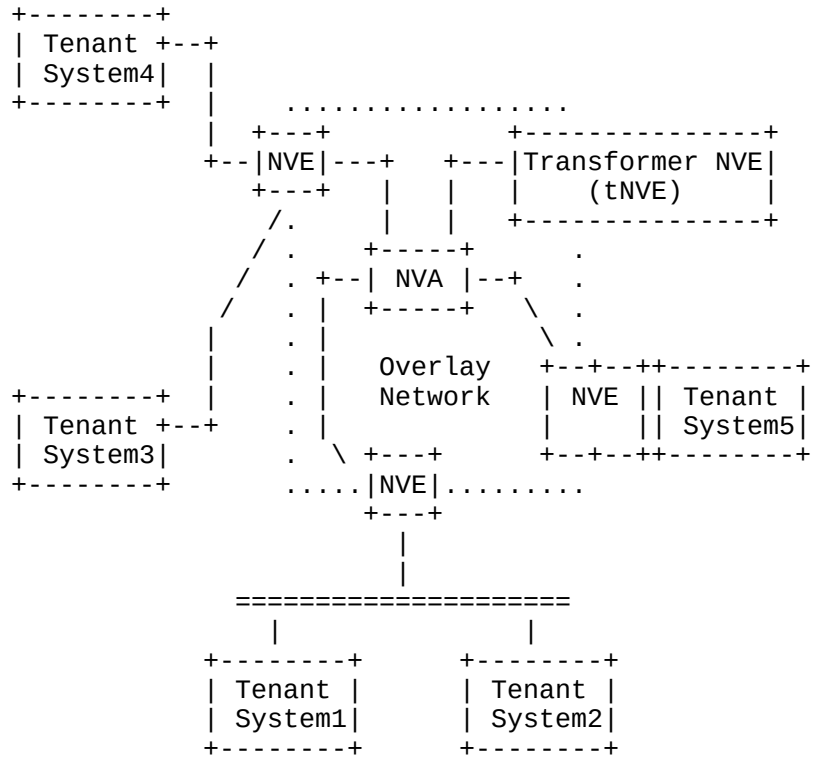



Figure 1 Multi-encapsulation VN architecture

Figure 1: Multi-encapsulation interconnection architecture

In this figure, a Transformer Gateway component is introduced. Generally, the gateway is located on a NVE, so we call it as tNVE. For the TSs in the same virtual network, if the NVE they connecting have different encapsulation, want to communicate with each other, Transformer NVE(tNVE) will take over as a gateway to provide a "bridge" for the communication. That is, when different NVEs want to set up tunnel, if they can’t connect each other directly because of different encapsulation, they can set up a tunnel with tNVE seperately, so that the tNVE connects the two tunnels as a transfer. There could be more than one tNVE in a network.

### 3.1. Transformer NVE

Transformer NVE(tNVE) is a certain kind of NVE that maybe appointed by NVA or by manager. As a very important component in the multi-encapsulation NVO, one requirement for NVE being a Transfomer NVE is that the NVE should support at least two kinds of encapsulations.

With reference to the [RFC8014], the Transformer NVE has a reference model as showed in Figure 2.

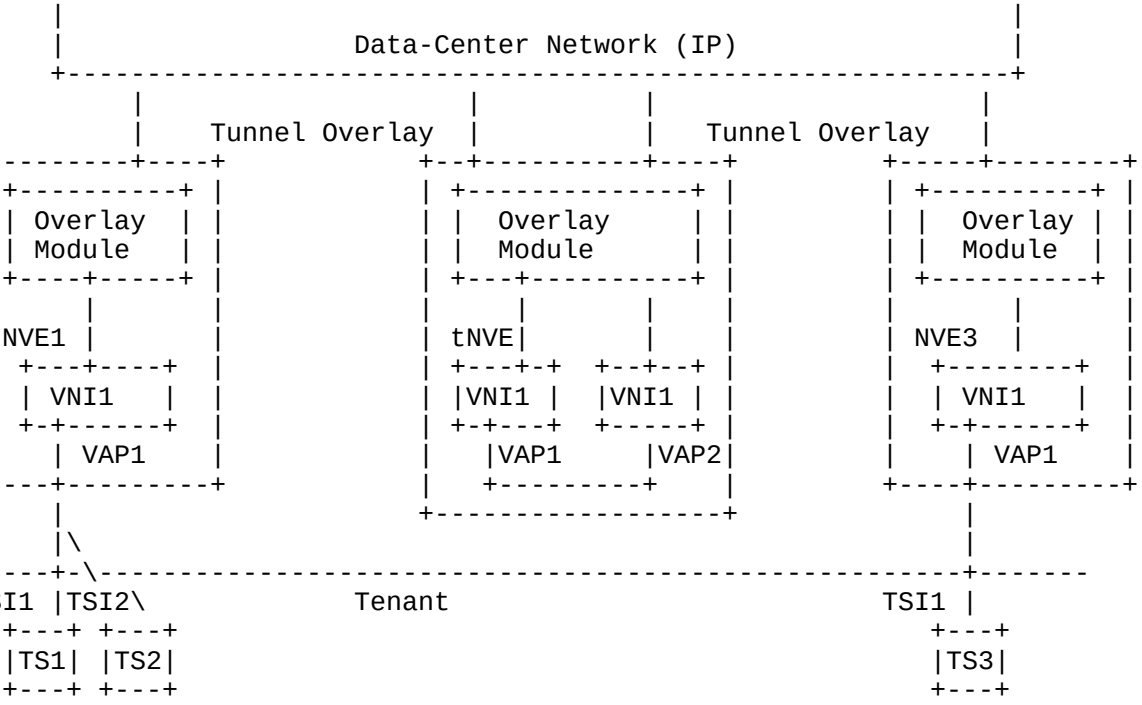



Figure 2 tNVE Reference Model

Figure 2: tNVE reference model

tNVE is a key component for the connection between NVE1 and NVE2. It can be a dedicated device and be a NVE that also provide the overlay network for the TSs. When the NVE takes the role of transform different encapsulation for different TSs, it will not forward the traffic to TS, but to another VAP that support the encapsulation the destination NVE owned.

Take the Figure 2 as an example to illustrate how does tNVE work. NVE1 only support VxLAN-GPE, and NVE2 only support GENEVE. For the two communiting TSs: TS1 wants to send packets to TS3, and TS3 also

wants to reply to TS1. They are in the same VNID1, but the NVE they are connecting using different encapsulation. So if the two TS communicate with each other, packets have to transfer at tNVE. For NVE1, it has no sense that TS3 is connecting to NVE3, instead assuming that TS3 is connecting to tNVE. In the same way, for NVE3, it has no sense that TS1 is connecting to NVE1, instead of assuming that TS1 is connecting to tNVE. So because of the existence of the tNVE, no matter TS1/TS3 or NVE1/NVE3, they never perceive that they are in different data plane. NVE1 getting the packets from TS1 encapsulates them in Vxlan-GPE and then send the packets to tNVE. The tNVE gets the packets from the Vxlan-GPE tunnel and then de-encapsulate the vxLAN-GPE to VAP1. Next the tNVE forward packets to the Overlay Module from VAP2 to have another encapsulation GENEVE on the packets. At last tNVE forward the packet in the GENEVE tunnel to NVE3.

From the above, tNVE is like a tranformer between TS1 and TS2. And owning to tNVE, even though NVE1 and NVE2 which TS1 and TS2 connecting seperately have different encapsulation, as long as they are in the same virtual network, they would communicate each other and no need to have knowledge that they are in different data plane.

## 4. Control Plane

As stated in [RFC8014], an NVA is the entity that provides address mapping and other information to NVEs. In addition, information flows between NVEs and NVAs in both directions. The NVA maintains information about all VNs in the NV Domain so that NVEs do not need to do so themselves. NVEs obtain information from the NVA about where a given remote TS destination resides. NVAs, in turn, obtain information from NVEs about the individual TSs attached to those NVEs.

Therefore, in order to make the tNVE works properly and to make sure that all the other network entities except tNVE don’t detect the difference, NVA should detect the role of tNVE and take the coordination role among tNVE and other NVEs, maintain the information about the tNVEs and other NVEs, compute the tunnel connection, and notify tNVEs and NVEs about remote TS information.

### 4.1. NVE to NVA

NVE and tNVE should not only notify the NVA the address mapping between NVE and TSs, but also notify the NVA which encapsulation tunnel does this mapping use, so that NVA be able to decide which connection need tNVE participation.

Information from tNVE to NVA:

1. Encapsulations the tNVE support.

Information from NVE to NVA:

1. The address mapping between NVE and its attached TSs.

2. The encapsulation tunnel the address mapping will use.

3. The mandate metadata the address mapping must use.

### 4.2. NVA to NVE

NVA notify the NVE and tNVE the mapping information after computing and coordination.

Information from NVA to NVE:

1. The address mapping information between remote NVE and TS. The NVE here may not be the NVE that the TS is connecting. It may be a tNVE.

Information from NVA to tNVE:

1. The address mapping information between remote NVE and its connecting TS.

### 4.3. NVA to NVA

For NVA federate scenario. To be added in the future updates.

## 5. Security Considerations

Will be added in the future updates.

## 6. IANA Considerations

TBD.

## 7. References

### 7.1. Normative References

[I-D.ietf-intarea-gue]
Herbert, T., Yong, L., and O. Zia, "Generic UDP Encapsulation", draft-ietf-intarea-gue-05 (work in progress), December 2017.

[I-D.ietf-nvo3-geneve]
Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-05 (work in progress), September 2017.

[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.

[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>.

[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>.

## 7.2. Informational References

[I-D.ietf-nvo3-encap]
Boutros, S., Ganga, I., Garg, P., Manur, R., Mizrahi, T., Mozes, D., Nordmark, E., Smith, M., Aldrin, S., and I. Bagdonas, "NVO3 Encapsulation Considerations", draft-ietf-nvo3-encap-01 (work in progress), October 2017.

[RFC7364] Narten, T., Ed., Gray, E., Ed., Black, D., Fang, L., Kreeger, L., and M. Napierala, "Problem Statement: Overlays for Network Virtualization", RFC 7364, DOI 10.17487/RFC7364, October 2014, <https://www.rfc-editor.org/info/rfc7364>.

Authors' Addresses

Ting Ao
ZTE Corporation
No.889, BiBo Road
Shanghai 201203
China

Phone: +86 21 68897642
Email: ao.ting@zte.com.cn

Greg Mirsky
ZTE Corp.
1900 McCarthy Blvd. #205
Milpitas, CA 95035
USA

Email: gregimirsky@gmail.com

Yongbin
China Telecom
No.109, Zhongshan Avenue
Guangzhou 510630
China

Email: fanyb@gsta.com

NVO3
Internet-Draft
Intended status: Informational
Expires: August 30, 2018

D. Migault
Ericsson
S. Boutros
D. Wing
VMware
S. Krishnan
Kaloom

February 26, 2018

# Geneve Protocol Security Requirements
# draft-mglt-nvo3-geneve-security-requirements-03

## Abstract

The document defines the security requirements to protect tenants overlay traffic against security threats from the NVO3 network components that are interconnected with tunnels implemented using Generic Network Virtualization Encapsulation (Geneve).

## 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 August 30, 2018.

Table of Contents



## 1. Requirements Notation

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 2. Introduction

The network virtualization overlay over Layer 3 (NVO3) as depicted in Figure 1, allows an overlay cloud provider to provide a logical L2/L3 interconnect for the Tenant Systems TSes that belong to a specific tenant network. A packet received from a TS is encapsulated by the ingress Network Virtualization Edge (NVE). The encapsulated packet is then sent to the remote NVE through a tunnel. When reaching the egress NVE of the tunnel, the packet is decapsulated and forwarded to the target TS. The L2/L3 address mappings to the remote NVE(s) are distributed to the NVEs by a logically centralized Network Virtualization Authority (NVA) or using a distributed control plane such as Ethernet-VPN. In a datacenter, the NVO3 tunnels can be implemented using Generic Network Virtualization Encapsulation (Geneve) [I-D.ietf-nvo3-geneve]. Such Geneve tunnels establish NVE-

to-NVE communications, may transit within the data center via Geneve Transit Nodes (GTN). The Geneve tunnels overlay network enable multiple Virtual Networks to coexist over a shared underlay infrastructure, and a Virtual Network may span a single data center or multiple data centers.

The underlay infrastructure on which the multi-tenancy overlay networks are hosted, can be owned and provided by an underlay provider who may be different from the overlay cloud provider.

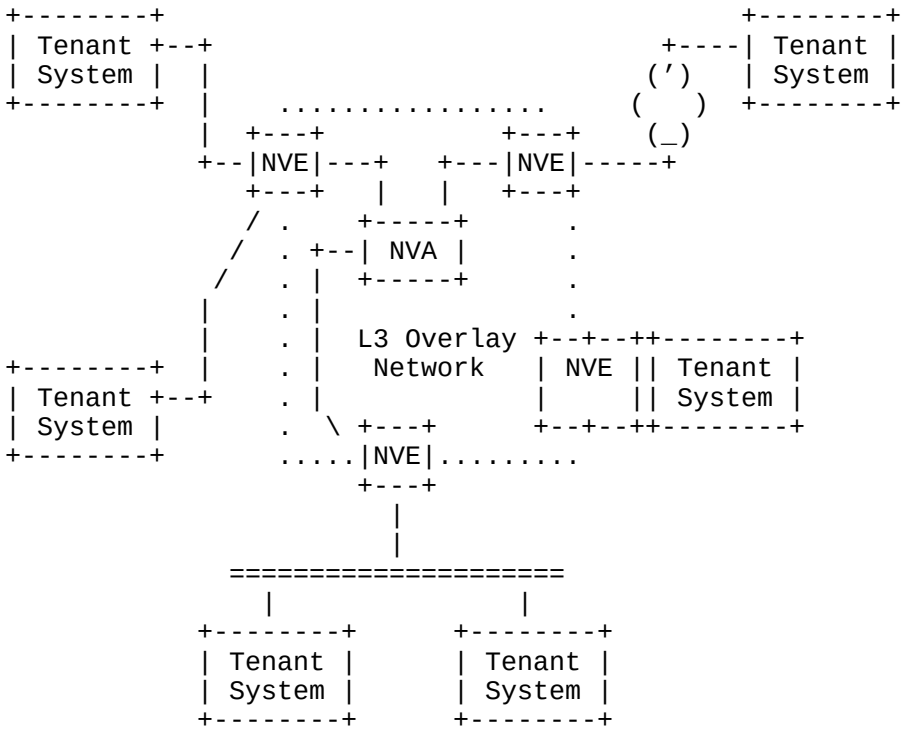



Figure 1: Generic Reference Model for Network Virtualization Overlays [RFC7365]

This document discusses the security risks that a Geneve based NVO3 network may encounter and tries to provide a list of essential security requirements that needs to be fulfilled. In addition, this document lists the requirements to protect the Geneve packet components defined in [I-D.ietf-nvo3-geneve] that include the Geneve tunnel IP and UDP header, the Geneve Header, Geneve options, and inner payload. Protecting the complete Geneve packet - that is the full IP packet or the full outer UDP payload for example - is out of

scope of this document, given that this can be supported using existing mechanisms.

This document assumes that a tenant subscribes to an overlay cloud provider for hosting its Tenant Systems, the cloud provider manages the Geneve overlay network on behalf of the tenant. The overlay network will be hosted on an underlay network infrastructure, that may be managed by another underlay cloud provider.

The security requirements in this document aims at providing the overlay cloud provider the necessary options to ensure:

1. Delivering tenant payload traffic and ensuring privacy and integrity of the overlay traffic, and isolation between the overlay and underlay networks, as well preventing tenant traffic from being redirected or injected to other tenants.

2. Protecting tenant traffic from rogue devices in the providers of Geneve overlay or underlay networks.

In summary, the document defines the security requirements to protect tenants overlay traffic against security threats from the NVO3 network components that are interconnected with tunnels implemented using Geneve.

The security requirements in the document are expressed regarding the threats to mitigate. It is expected that a security mechanism designed for NVO3 overlay network implementing Geneve is able to mitigate all the threats and as such fulfills all the security requirements expressed in the document. The document RECOMMENDs that the definition of a Geneve security mechanism fulfills all requirements expressed in this document

On the other hand, the specificities and the context of some Geneve deployments may consider the risk associated to some threats as very low and as such ignore the threats. In such cases, a specific security mechanism designed for that specific deployment may not fulfill all the requirements associated to that given threat. This document RECOMMENDs to consider all the threats while designing a security mechanism for the Geneve overlay network. In addition, some deployments may not take advantage of some features provided by Geneve, in which case, a specific security mechanisms designed for a specific deployment may not fulfill the requirements associated to that feature. This document RECOMMENDs that such specific security mechanisms be an intermediary approach toward the deployment of a Geneve security mechanism. In fact, such specific mechanisms present the risk to ossifying Geneve as well as the security being lowered in favor of Geneve features.

The document strongly recommend to re-use existing security protocols like IP Security (IPsec) [RFC4301] and Transport Layer Security (TLS) [RFC5246], and existing encryption algorithms ( such as [RFC8221]), and authentication protocols.

Authentication requirements for NVO3 devices, automated key management, as well as packet level security providing confidentiality, integrity and authorization requirements defined in [I-D.ietf-nvo3-security-requirements] are also requirements for this document.

## 3. Terminology

This document uses the terminology of [RFC8014], [RFC7365] and [I-D.ietf-nvo3-geneve]. In addition to these document the following term is used:

- Immutable Geneve Option: designate a Geneve Option that are not expected to be modified by any on path element, such as a GTN.

- Geneve Transit Node (GTN): A transit device that is not Geneve termination point. GTN MAY understand and Geneve packet and MAY process Geneve Option.

## 4. Security Threats

Attacks from compromised NVO3 and underlay network devices, and attacks from compromised tenant systems defined in [I-D.ietf-nvo3-security-requirements] are considered for the Geneve overlay network. Furthermore, the attackers knowing the details of the Geneve packets can perform their attacks by changing fields in the Geneve tunnel header, base header, Geneve options and Geneve packet inner payload.

Threats include traffic analysis, sniffing, injection, redirection, and replay. Based on these threats, this document enumerates the security requirements.

Threats are divided into two categories: passive attack and active attack.

Threats are always associated with risks and the evaluation of these risks depend among other things on the environment.

### 4.1. Passive Attacks

Passive attacks include traffic analysis (noticing which workloads are communicating with which other workloads, how much traffic, and when those communications occur) and sniffing (examining traffic for useful information such as personally-identifyable information or protocol information (e.g., TLS certificate, overlay routing protocols).

A rogue element of the overlay Geneve network under the control of an attacker may leak and redirect the traffic from a virtual network to the attacker for passive monitoring [RFC7258].

Avoiding leaking information is hard to enforced and the security requirements expect to mitigate such attacks by lowering the consequences, typically making leaked data unusable to an attacker..

### 4.2. Active Attacks

Active attacks involve modifying packets, injecting packets, or interfering with packet delivery (such as by corrupting packet checksum).

There are multiple motivations to inject illegitimate traffic into a tenants network. When the rogue element is on the path of the TS traffic, it may be able to inject and receive the corresponding messages back. On the other hand, if the attacker is not on the path of the TS traffic it may be limited to only inject traffic to a TS without receiving any response back. When rogue element have access to the traffic in both directions, the possibilities are only limited by the capabilities of the other on path elements - GTN, NVE or TS - to detect and protect against the illegitimate traffic. On the other hand, when the rogue element is not on path, the surface for such attacks remains still quite large. For example, an attacker may target a specific TS or application by crafting a specific packet that can either generate load on the system or crash the system or application. TCP syn flood typically overload the TS while not requiring the ability to receive responses. Note that udp application are privileged target as they do not require the establishment of a session and are expected to treat any incoming packets.

Traffic injection may also be used to flood the virtual network to disrupt the communications between the TS or to introduce additional cost for the tenant, for example when pricing considers the traffic inside the virtual network. The two latest attacks may also take advantage of applications with a large factor of amplification for their responses as well as applications that upon receiving a packet

interact with multiple TS. Similarly, applications running on top of UDP are privileged targets.

Note also that an attacker that is not able to receive the response traffic, may use other channels to evaluate or measure the impact of the attack. Typically, in the case of a service, the attacker may have access, for example, to a user interface that provides indication on the level of disruption and the success of an attack, Such feed backs may also be used by the attacker to discover or scan the network.

Preventing traffic to cross virtual networks, reduce the surface of attack, but rogue element main still perform attacks within a given virtual network by replaying a legitimate packet. Some variant of such attack also includes modification of unprotected parts when available in order for example to increase the payload size.

## 5. Requirements for Security Mitigations

The document assumes that Security protocols, algorithms, and implementations provide the security properties for which they are designed, an attack caused by a weakness in a cryptographic algorithm is out of scope.

Protecting network connecting TSes and NVEs which could be accessible to outside attackers is out of scope.

An attacker controlling an underlying network device may break the communication of the overlays by discarding or delaying the delivery of the packets passing through it. The security consideration to prevent this type of attack is out of scope of this document.

Securing communication between NVAs and NVEs is out of scope.

Selectively providing integrity / authentication, confidentiality / encryption of only portions of the Geneve packet is in scope. This will be the case if the Tenant Systems uses security protocol to protect its communications.

### 5.1. Protection Against Traffic Sniffing

A passive network observer can determine two virtual machines are communicating by manipulating activity or network activity of other virtual machines on that same host. For example, the attacker could control (or be otherwise aware of) network activity of the other VMs running on the same host, and deduce other network activity is due to a victim VM. Comparing application TLS to guest IPsec ESP to NVE IPsec ESP, each provides stronger protection from traffic analysis in

the same order. Application TLS exposes TCP port numbers to a passive observer, guest IPsec ESP encrypts the inner transport header but still identifies the communicating VM's IP address, while NVE IPsec ESP encrypts the entire inner payload.

To protect packet payloads from passive listeners, application-level encryption (e.g., JSON Web Encryption [RFC7516]), application TLS, guest IPsec ESP, or hypervisor IPsec ESP can be used. Each provides the same protection against a passive listener.

To protect against the above-described traffic sniffing attacks, we require:

GEN-REQ1: The NVE MUST ensure the traffic leaving the NVE has its payload encrypted. The encryption operation MAY be performed by the NVE, but could also be performed, for example, by the TS.

GEN-REQ2: To provide best protection from traffic analysis, the NVE SHOULD encrypt the payload fields that appears in clear. Typically, this could include VM's inner IP address, transport header, and IP payload when Geneve carries IP packets.

GEN-REQ1 and GEN-REQ2 are inline with NVE-NVE and NVE-Hypervisor data plane security requirements for confidentiality in [I-D.ietf-nvo3-security-requirements] like REQ 10, 11 and 16.

## 5.2. Protecting Against Traffic Injection

Traffic injection from a rogue non legitimate NVO3 Geneve overlay device or a rogue underlay transit device can target an NVE, a transit underlay device or a Tenant System. Targeting a Tenant's System requires a valid MAC and IP addresses of the Tenant's System.

Tenant's System may protect their communications using IPsec or TLS. Such protection protects the Tenants from receiving spoofed packets, as any injected packet is expected to be discarded by the destination Tenant's System. Such protection does not protect the tenant system from receiving illegitimate packets that may disrupt the Tenant's System performance.

The Geneve overlay network MAY still need to prevent such spoofed Tenant's system packets from being steered to the Tenant's system.

When the Tenant's System are not protecting their communications, the Geneve overlay network SHOULD be able to to prevent a rogue device from injecting traffic into the overlay network.

In order to prevent traffic injection to one virtual network, the destination legitimate Geneve NVE MUST be able to authenticate the incoming Geneve packets from the source NVE. Authenticated Geneve Packet MAY be checked by underlay intermediary nodes.

Based on a policy partial authentication MAY be performed on Geneve packets if tenant's system is protecting it's communication. In situations where the tenant system is already encrypting its traffic with application-level encryption (e.g., S/MIME), transport encryption (e.g., TLS), or IP encryption (e.g., IPsec ESP), it is redundant for the NVE to apply additional encryption. Note that relaying on upper security layers, results from a compromise between security and performance as it may introduce cut and paste vulnerability.

The Geneve architecture considers intermediary nodes designated as GTN. A protection established between NVE SHOULD NOT prevent GTN to perform their operations, such as the insertion of a Geneve Option, authenticating a Geneve Option or steering Geneve packets. In the later case, in order to ease the transition from a non secured to secure Geneve overlay network, it is expected that GTN that are not aware of Geneve security mechanisms can steer authenticated Geneve packets the same way as non protected Geneve packets. Similarly, the transition from non secure to secure Geneve overlay network may also be performed by introducing GTN that performs the security functionalities - such as authentication of Geneve packets- on behalf of NVE.

This leads to the following security requirements:

GEN-REQ3: A Geneve NVE MUST be able to authenticate at least one of the Geneve tunnel Header, the Geneve base header, the immutable Geneve Options, or the Geneve payload. The combination of fields that are authenticated is defined by security policies.

GEN-REQ4: A Geneve NVE MAY be able to authenticate only a portion of the Geneve payload if the Tenant's system is protecting its communication.

GEN-REQ5: A GTN MAY be able to validate the authentication before the packet reaches the Geneve destination NVE.

GEN-REQ6: A GTN MUST be able to insert an authenticated Geneve Option into a authenticated Geneve Packet - protected by the source Geneve NVE.

GEN-REQ7: A GTN capable of forwarding non-authenticated Geneve packets MUST be capable of forwarding the Geneve authenticated packet without any additional security specific functionalities. In other words, forwarding authenticated Geneve packet MUST done the same way as authenticated Geneve packets.

GEN-REQ8: A Geneve NVE SHOULD be able to set different security policies for different flows. A flow MUST be identified at minimum by the Geneve virtual network identifier and the inner IP and transport headers, and optionally additional fields which define a flow (e.g., inner IP DSCP, IPv6 flow id, Geneve options).

GEN-REQ9: In the case when Tenant systems secure their communications using protocols such as TLS or IPsec. A Geneve NVE MAY be able to selectively encrypt and/or authenticate only the sections that are not encrypted/ authenticated by the Tenant System. For example, only the IP, transport (TCP / UDP) in case of TLS/DTLS MAY be encrypted/authenticated, while only the IP header and ESP header MAY be encrypted/authenticated.

Requirements listed in this section are inline with authentication and integrity requirements in [I-D.ietf-nvo3-security-requirements], like REQ 9, 10, 11, 14 and 16.

The requirements further define mechanisms to fully and partially authenticate Geneve Header, and Geneve options, as well fully and partially encrypt the same.

### 5.3. Protecting Against Traffic Redirection

A rogue device of the NVO3 overlay Geneve network or the underlay network may redirect the traffic from a virtual network to the attacker for passive or active attacks. If the rogue device is in charge of the securing the Geneve packet, then Geneve security mechanisms are not intended to address this threat. More specifically, a rogue source NVE will still be able to redirect the traffic in clear text before protecting ( and encrypting the packet). A rogue destination NVE will still be able to redirect the traffic in clear text after decrypting the Geneve packets. The same occurs with GTN that are in charge of encrypting and decrypting a Geneve Packet, Geneve Option or any information. The security mechanisms are intended to protect a Geneve information from any on path node.

To prevent an attacker located in the middle between the NVEs and modifying the tunnel address information in the data packet header to

redirect the data traffic, the solution need to provide confidentiality protection for data traffics exchanged between NVEs.

Based on a policy partial encryption MAY be performed on Geneve packets if tenant's system is protecting it's communication.

The Geneve architecture considers intermediary nodes designated as GTN. A protection established between NVE SHOULD NOT prevent GTN to perform their operations, such as the insertion of a Geneve Option, encrypting a Geneve Option or steering Geneve packets. In the later case, in order to ease the transition from a non secured to secure Geneve overlay network, it is expected that GTN that are not aware of Geneve security mechanisms can steer encrypted Geneve packets the same way as non protected Geneve packets. Similarly, the transition from non secure to secure Geneve overlay network may also be performed by introducing GTN that performs the security functionalities - such as encryption of Geneve packets- on behalf of NVE.

This leads to the following security requirements:

GEN-REQ10: A Geneve NVE MUST be able encrypt Geneve base Header, and/ or Geneve Payload and/or Geneve Options not intended for the GTN.

GEN-REQ11: A Geneve NVE MAY be able encrypt portion of Geneve Payload as well as as Geneve Options not intended for the GTN.

GEN-REQ12: A transit underlay intermediary node MUST be able to insert an encrypted Geneve Option into an encrypted/ authenticated Geneve Packet - protected by the source Geneve NVE.

GEN-REQ13: A Geneve NVE SHOULD be able to assign different cryptographic keys to protect the unicast tunnels between NVEs respectively.

GEN-REQ14: If there are multicast packets, a Geneve NVE SHOULD be able to assign distinct cryptographic group keys to protect the multicast packets exchanged among the NVEs within different multicast groups. Upon receiving a data packet, an egress Geneve NVE MUST be able to verify whether the packet is sent from a proper ingress NVE which is authorized to forward that packet.

Requirements listed in this section are inline with the requirements in the data plane sections in [I-D.ietf-nvo3-security-requirements] to protect against traffic redirection and man in the middle attacks.

The requirements further define mechanisms for a transit intermediary node to insert an encrypted Geneve option to an encrypted/ authenticated Geneve packet.

### 5.4. Protecting Against Traffic Replay

A rogue device of the NVO3 overlay Geneve network or the underlay network may replay a Geneve packet, to load the network and/or a specific Tenant System with a modified Geneve payload. In some cases, such attacks may target an increase of the tenants costs.

When traffic between tenants is not protected, the rogue device may forward the modified packet over a valid Geneve Header. The crafted packet may for example, include a specifically crafted application payload for a specific Tenant Systems application, with the intention to load the tenant specific application.

Updating the Geneve header and option parameters such as setting an OAM bit, adding bogus option TLVs, or setting a critical bit, may result in different processing behavior, that could greatly impact performance of the overlay network and the underlay infrastructure and thus affect the tenants traffic delivery.

The NVO3 overlay network and underlay network nodes that may address such attacks MUST provide means to authenticate the Geneve packet components.

This leads to the following security requirements:

GEN-REQ15: A Geneve NVE or a GTN MUST be able to validate the Geneve Header corresponds to the Geneve payload, and discard such packets.

GEN-REQ16: A Geneve NVE or a GTN SHOULD provide anti replay mechanisms and discard replayed packet.

The requirements in this section are inline with REQ 10 and 14 in [I-D.ietf-nvo3-security-requirements], and they further specifies requirements to validate that a Geneve Header corresponds to the Geneve payload.

## 6. IANA Considerations

There are no IANA consideration for this document.

## 7. Security Considerations

The whole document is about security.

Limiting the coverage of the authentication / encryption provides some means for an attack to craft special packets.

The current document details security requirements that are related to the Geneve protocol. Their purpose is to design appropriated Geneve security Options or to appropriately secure NVE-NVE communication based on Geneve. Instead, [I-D.ietf-nvo3-security-requirements] provides generic architecture security requirement upon the deployment of an NVO3 overlay network. It is strongly recommended to read that document as architecture requirements also apply here. In addition, architecture security requirements go beyond the scope of Geneve communications, and as such are more likely to adress the security needs upon deploying an Geneve overlay network.

More precisely, REQ 1 to REQ 8 are focused on the control plane which is outside the scope of this document.

REQ 9 is a data plane security requirement, but focused on the establishment of a NVO3 tunnels. This is outside the scope of Geneve which only address data in motion. As such REQ 9 is outside the scope of this document.

REQ 10 to REQ 14 are in the scope of this document. REQ 12 and REQ 13 are identical as GEN-REQ13 and GEN-REQ14. All other requirements from GEN-REQ1 to GEN-REQ16 are declinasons of REQ 10, REQ 11 and REQ 14. These requirements are the declination of architecture requirements in a context for Geneve, which includes the presence of GTN, Geneve Options as well as the possibility to split the security opration between tenants and teh overlay infrastructure.

REQ 15 to REQ 18 from [I-D.ietf-nvo3-security-requirements] are focused on the NVE-Hypervisor Data Plane which is not based on Geneve and thus is outside the scope of the document.

## 8. Acknowledgments

We would like to thank Ilango S Ganaga for its useful reviews and clarifications as well as Matthew Bocci, Sam Aldrin and Ignas Bagdona for moving the work forward.

# 9. References

## 9.1. Normative References

[I-D.ietf-nvo3-geneve]
Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-05 (work in progress), September 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>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[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>.

[RFC8221] Wouters, P., Migault, D., Mattsson, J., Nir, Y., and T. Kivinen, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 8221, DOI 10.17487/RFC8221, October 2017, <https://www.rfc-editor.org/info/rfc8221>.

## 9.2. Informational References

[I-D.ietf-nvo3-security-requirements]
Hartman, S., Zhang, D., Wasserman, M., Qiang, Z., and M. Zhang, "Security Requirements of NVO3", draft-ietf-nvo3-security-requirements-07 (work in progress), June 2016.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <https://www.rfc-editor.org/info/rfc7258>.

[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>.

[RFC7516] Jones, M. and J. Hildebrand, "JSON Web Encryption (JWE)", RFC 7516, DOI 10.17487/RFC7516, May 2015, <https://www.rfc-editor.org/info/rfc7516>.

[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>.

## Authors' Addresses

Daniel Migault
Ericsson
8400 boulevard Decarie
Montreal, QC  H4P 2N2
Canada

Email: daniel.migault@ericsson.com

Sami Boutros
VMware, Inc.

Email: sboutros@vmware.com

Dan Wing
VMware, Inc.

Email: dwing@vmware.com

Suresh Krishnan
Kaloom

Email: suresh@kaloom.com

NVO3 Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 11, 2017

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

March 10, 2017

# OAM Header for use in Overlay Networks
draft-ooamdt-rtgwg-ooam-header-03

## 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

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 September 11, 2017.

## 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 [I-D.ietf-bier-mpls-encapsulation], and NSH [I-D.ietf-sfc-nsh] 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:

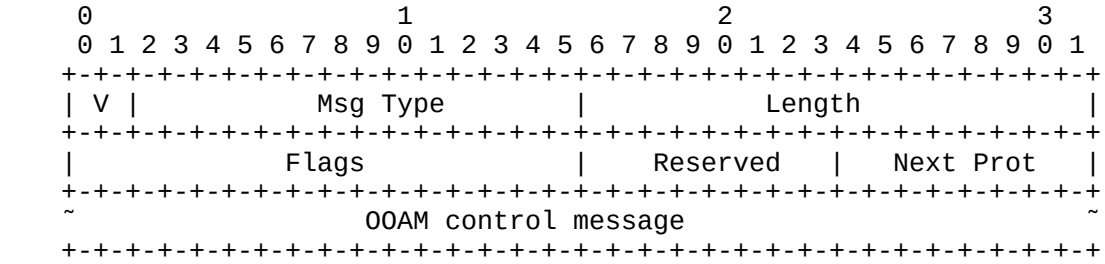



Figure 1: Overlay OAM Header format

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:

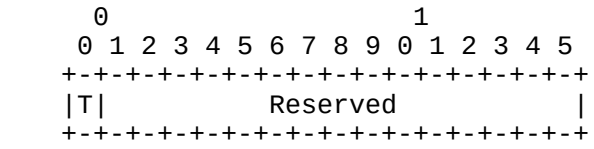



Figure 2: Flags field format

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.

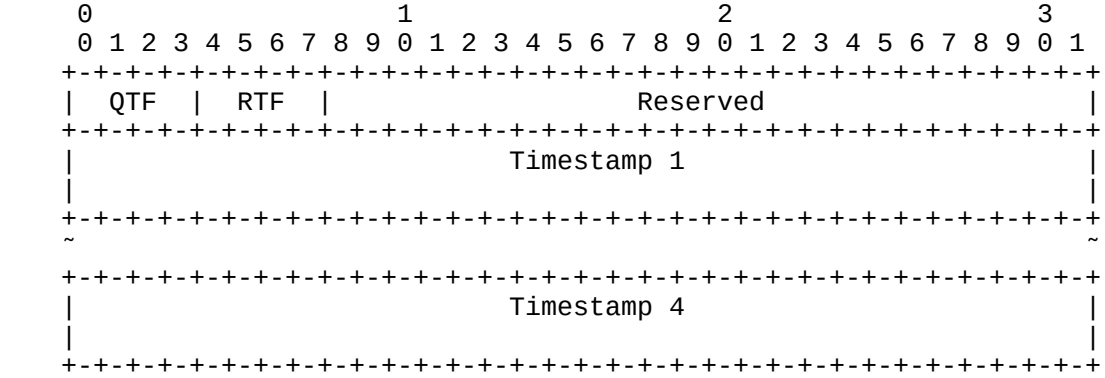



Figure 3: Timestamp block format

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 [I-D.ietf-ippm-twamp-time-format]. 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.

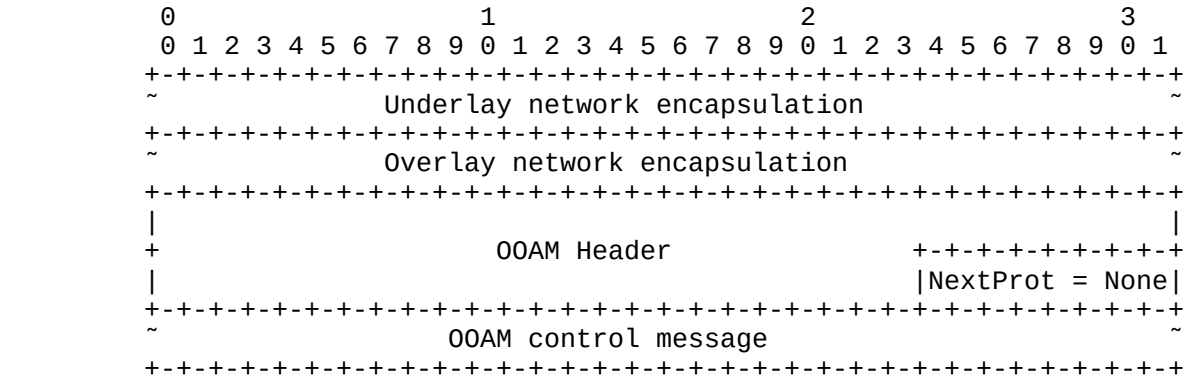



Figure 4: Overlay OAM Header in Active OAM Control Packet

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.

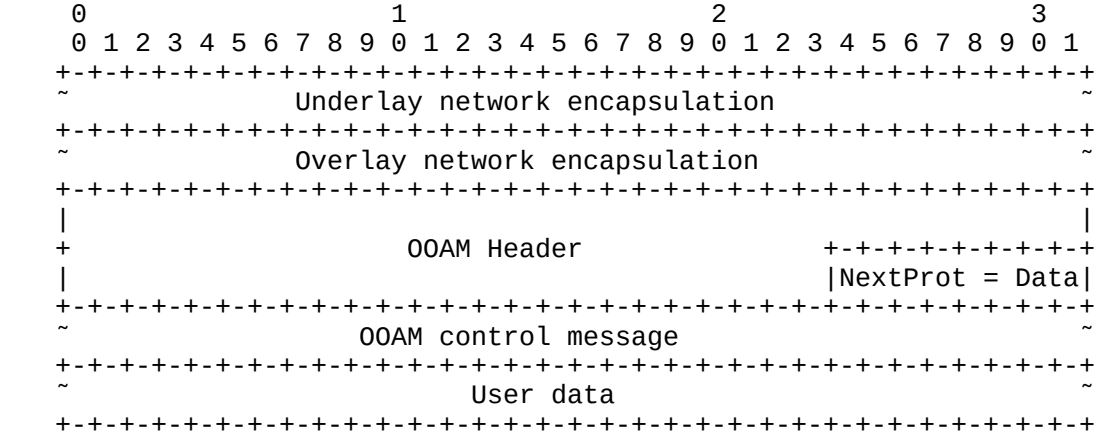



Figure 5: Overlay OAM Header in Hybrid OAM Control Packet

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 [RFC5226] . 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 [RFC5226].

| 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, <http://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, <http://www.rfc-editor.org/info/rfc5905>.

### 9.2. Informative References

[I-D.ietf-bier-mpls-encapsulation]
Wijnands, I., Rosen, E., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication in MPLS and non-MPLS Networks", draft-ietf-bier-mpls-encapsulation-06 (work in progress), December 2016.

[I-D.ietf-ippm-twamp-time-format]
Mirsky, G. and I. Meilik, "Support of IEEE-1588 time stamp format in Two-Way Active Measurement Protocol (TWAMP)", draft-ietf-ippm-twamp-time-format-05 (work in progress), March 2017.

[I-D.ietf-nvo3-geneve]
Gross, J., Ganga, I., and T. Sridhar, "Geneve: Generic Network Virtualization Encapsulation", draft-ietf-nvo3-geneve-03 (work in progress), September 2016.

[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-03 (work in progress), October 2016.

[I-D.ietf-sfc-nsh]
Quinn, P. and U. Elzur, "Network Service Header", draft-ietf-sfc-nsh-12 (work in progress), February 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, <http://www.rfc-editor.org/info/rfc4656>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[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, <http://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, <http://www.rfc-editor.org/info/rfc7799>.

## 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

INTERNET-DRAFT
Intended Status: Standards Track
Expires: Sep 1, 2018

H. Xiang
Y. Yu
Huawei Technologies
P. Congdon
Tallac Networks
J. Wang
China Telecom
March 1, 2018

# Packet Spraying in Geneve Overlay Network
draft-xiang-nvo3-geneve-packet-spray-00

## Abstract

Congestion is the killer of low latency and high throughput.Network congestion occurs on the interconnection links of a data center due to poor traffic distribution. Load balancing technologies are used to solve network congestion. Packet spraying is a kind of load balancing technology with finer granularity. This document describes a packet spraying protocol in the Geneve encapsulation network[1] using a newly defined Geneve Option field.

## 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/1id-abstracts.html

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

## Copyright and License Notice

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

In many current data centers, network utilization is not has high as it could be. For example, in some scenarios, the average network utilization is about 20% and the peak utilization is about 45%[2]. With the improvement of end systems (or endpoints), the deployment of multi-services and high-volume traffic services (such as streaming media, big data processing applications and user-oriented large-scale web applications, etc.), more and more network performance problems appear. These problems are created by traffic bursts and traffic routing collisions. The imbalance of traffic on the network becomes more and more prominent which leads to underutilized network bandwidth and decreased overall performance of network applications.

In order to fully utilize the available network bandwidth, traffic flows into the network are dispersed across multiple paths to achieve load balancing. The finer the granularity of the load balancing, the higher the utilization of available network bandwidth. Current flow-based and flowlet-based[3] approaches are more coarse grain than packet-based load balancing. This document describes how to extend

the Geneve header to support packet-based load balancing, called packet spraying in the Geneve encapsulation network.

## 2 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].

## 3 Abbreviations

GENEVE - Generic Network Virtualization Encapsulation

ECMP - Equal-cost multi-path routing

SDN - Software Defined Network

GFP - Geneve Forwarding Policy

## 4 Problem Statements & Requirements

The current general network topology in the data center is a multi-rooted tree architecture, such as the typical CLOS network. This kind of network has multiple paths and an equal division of bandwidth across those paths which provides good scalability and flexibility depending on how the multiple paths are utilized. In order to fully utilize the network bandwidth, traffic flows into the network are dispersed on the multiple paths to achieve load balancing. Currently,



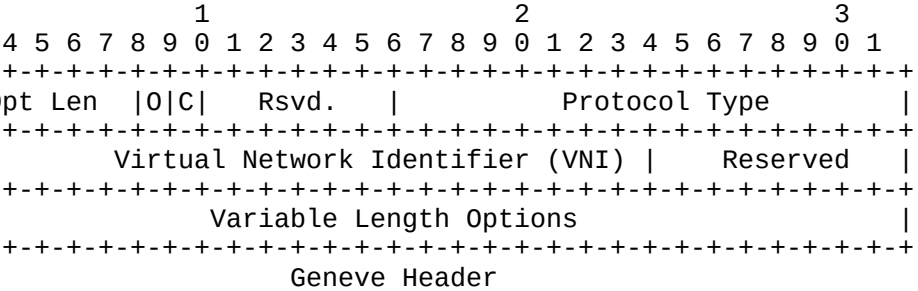

the granularity of load balancing can be seen in the following approaches: flow-based load balancing (such as ECMP), flowlet-based load balancing (such as CONGA[2]) and packet-based load balancing (such as Packet Spraying). The finer the granularity of load balancing, the more effective the load balancing is and the higher the utilization of network bandwidth can be.

The effect of packet-based load balancing is the best one among the three because the corresponding granularity is the smallest. However, the consequence is that packets belonging to the same flow will be allocated to different paths. When the forwarding delays of paths are different, it is possible that packets may arrive at the receiver out-of-order. To detect out-of-order packets and restore the correct order, a sequence number is needed in the packets.

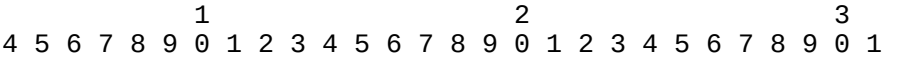
## 5 Packet Spraying on Geneve

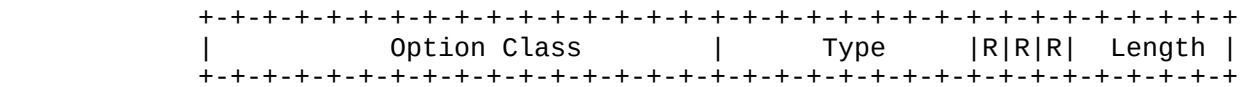
### 5.1 Packet Spraying Format

The Geneve Header and the Geneve option have the following format[1]:




Geneve Header



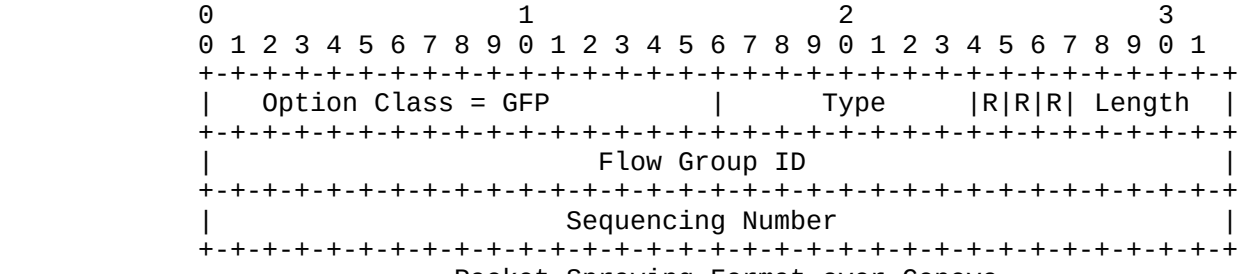



Geneve Option

Option Class = To be assigned by IANA (TBA).
Type = TBA.
Length = 2 (8 byte)

The proposed Packet Spraying option for Geneve will have the following format:






Packet Spraying Format over Geneve

Option Class = Geneve Forwarding Policy(suggested), to be assigned by IANA (TBA).
Type = TBA.
Length = 2 (8 byte)

Flow Group ID: will be described in 5.1.1

Sequencing Number: will be described in 5.1.2

### 5.1.1 Flow Group ID Field (4 Bytes)

The Flow Group ID field is a four byte field. The Flow Group ID identifies a group of flows within the same reorder sequence space between a pair of src/dest nodes. The Flow Group ID may correspond to an individual flow, some subset of flows, or even all flows between the src/dest pair. How the flow corresponds to the Flow Group ID is not defined by this draft. The same Flow Group ID can be used by different src/dest pairs (i.e. a Flow Group ID is only unique within the context of a src/dest pair). A Flow Group is uniquely identified by the 3 tuple that includes src IP, dest IP and Flow Group ID. The source node allocates the sequence number according to the order packets are sent for flows of the same Flow Group. The destination will reorder the received packets of a Flow Group according to the received sequence number.

### 5.1.2 Sequence Number Field

The Sequence Number field is a four byte field that closely follows the definition of the Sequence Number in RFC 2890[4]. The sequence number value ranges from 0 to (2**32)-1. The first datagram is sent with a sequence number of 0. The sequence number is thus a monotonically increasing counter represented modulo 2**32. The receiver maintains the sequence number value of the last successfully

decapsulated packet. This value should be initialized to (2**32)-1.

A packet is considered an out-of-sequence packet if the sequence number of the received packet is less than or equal to the sequence



number of last successfully decapsulated packet. The sequence number of a received message is considered less than or equal to the last successfully received sequence number if its value lies in the range of the last received sequence number and the preceding 2**31-1 values, inclusive.

If the received packet is an in-sequence packet, it is successfully decapsulated. An in-sequence packet is one with a sequence number exactly 1 greater than (modulo 2**32) the last successfully decapsulated packet. If the received packet is neither an in-sequence nor an out-of-sequence packet it indicates a sequence number gap. The receiver may perform a small amount of buffering in an attempt to recover the original sequence of transmitted packets. In this case, the packet may be placed in a buffer sorted by sequence number. If an in-sequence packet is received and successfully decapsulated, the receiver should consult the head of this buffer to see if the next in-sequence packet has already been received. If so, the receiver should decapsulate it as well as the following in-sequence packets that may be present in the buffer. The "last successfully decapsulated sequence number" should then be set to the last packet that was decapsulated from the buffer.

Under no circumstances should a packet wait more that OUTOFORDER_TIMER milliseconds in the buffer. If a packet has been waiting that long, the receiver MUST immediately traverse the buffer in sorted order, decapsulating packets (and ignoring any sequence number gaps) until there are no more packets in the buffer that have been waiting longer than OUTOFORDER_TIMER milliseconds. The "last successfully decapsulated sequence number" should then be set to the last packet so decapsulated.

The receiver may place a limit on the number of packets in any per-flow group buffer (Packets with the same Flow Group ID Field value belong to a flow group). If a packet arrives that would cause the receiver to place more than MAX_PERFLOW_BUFFER packets into a given buffer, then the packet at the head of the buffer is immediately decapsulated regardless of its sequence number and the "last successfully decapsulated sequence number" is set to its sequence number. The newly arrived packet may then be placed in the buffer.

The received packets of flows from the same Flow Group are in the same reorder sequence space. The source ensures to allocate the sequence number according to the sequence of sent packets. If the sequence number wraps, the source will allocate from 0 again.

## 5.2 Packet Spray Capability Discovery



The reorder function on the destination needs certain resources. For

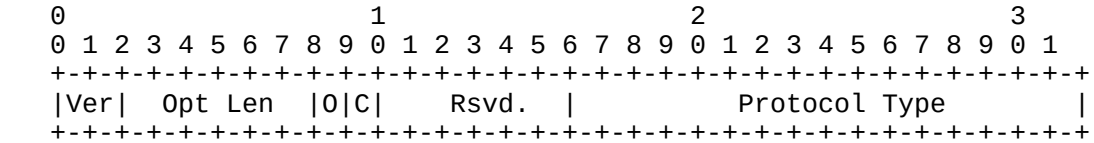
example, there is a reorder queue corresponding to each Group ID(Flow Group ID plus the Source IP address). For some resource-intensive chips such as switch chips, the amount of queues are limited. Therefore, it is important to not exceed the ability of the destination when assigning the Group ID at the source. This requires that the source understands the ability of the destination. There are several solutions, such as static configuration, or direct signaling between the two ends. In the following situations, the capability notifications need to be sent to the peer:
1. When the source communicates with the destination for the first time.
2. When receiving the peer packet for the first time
3. When receiving the capability notification from the source
4. When the Group ID of peer exceed the local capability

In the above cases, the destination needs to notify the capability (reorder queues assigned to the peer) to the source. When receiving the capability notification from the destination, the source needs to tune the allocation mechanism of Group ID according to the capability of destination to ensure the number of Group IDs does not exceed the number of reordering queues allocated to the source.

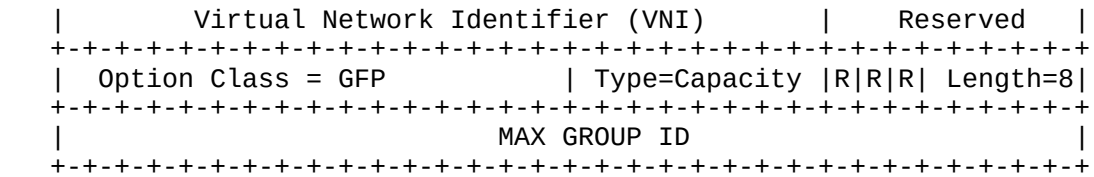
When the number of Group IDs exceed the local capability, the following 2 actions can be taken. Which option is selected is not covered in this draft.
1.Discard the Geneve packet for the Group ID that exceeds the local capability

2.Remove the Geneve encapsulation, without performing reordering and pass the packet to higher layer protocol. For higher layer protocols that can tolerate a certain degree of out-of-order packets (such as TCP), the message may be processed correctly.

When the Group ID exceeds the local capability, the destination sends a notification of the reordering capability to the source. To prevent sending the capability notification too frequently, a notification suppression capability is needed. When the destination wants to send a notification of the capability of the source, it enters a suppression cycle. The destination will not send the capability notification to the source until the suppression cycle ends. The suppression period is longer than the RTT between 2 nodes.






Capability notification message format

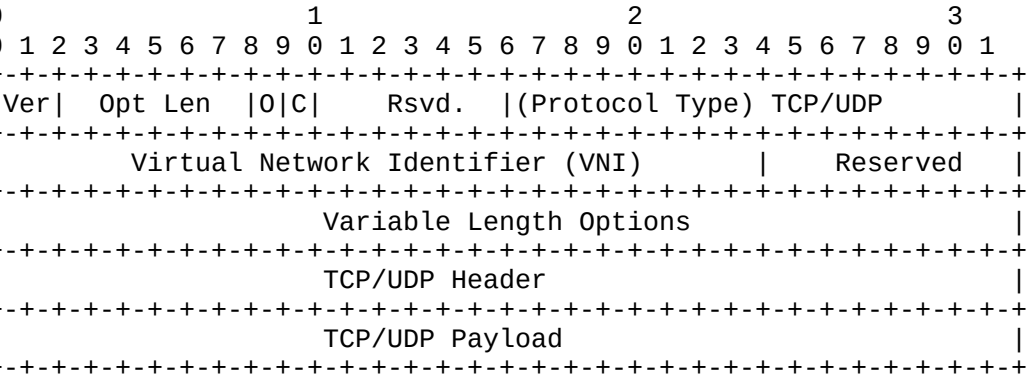
MAX GROUP ID is a four byte field. MAX Group ID indicate the max Group ID assigned to the destination. The Group ID allocated by the source must be limited to 0 ~ MAX Group ID.

## 5.3 TCP/UDP over Geneve

For some certain applications, the main parts of outer IP header are the same with Geneve inner IP header. For example, source/destination nodes and IP addresses are the same on both inner and outer header. When source/destination nodes are same, TCP/UDP layer could be over the Geneve directly, saving 20 bytes(IPv4) in the header. In this situation, the Geneve header Protocol Type must specify the transport layer protocol.

When the destination receives such a message, it strips off the Geneve header directly and splices the TCP/UDP message to the back of the IP header.

Geneve Header:




## 6 Security Considerations

This document describes Geneve option which introduce Flow Group ID and Sequence Number to reorder packets. Within the Sequence Number Field, it is possible to inject packets with an arbitrary Sequence Number and launch a Denial of Service attack. This is a general



security issue which is defined in Geneve security requirements[5].

In order to protect against such attacks, IPSec could be used to protect the Geneve header and the tunneled payload. Any common Geneve security mechanism also applies to this draft.

## 7 IANA Considerations

IANA is requested to allocate a Geneve "option class" number for GFP(Geneve Forwarding Policy):

| Option Class | Description | Reference |
|---|---|---|
| x | GFP_ID | This document |

IANA/IEEE is requested to allocate a Geneve "Protocol Type" number for TCP/UDP over Geneve:

| Protocol Type | Description | Reference |
|---|---|---|
| 0x9004 | TCP | This document |

| 0x9005 | UDP | This document |
+---+---+---+

## 8 References


[1] J. Gross, Ed., I. Ganga, Ed., T. Sridhar, Ed., "Generic Network Virtualization Encapsulation", [I-D.ietf-nvo3-geneve]

[2] Jiaxin Cao, et al, "Per-packet Load-balanced, Low-Latency Routing for Clos-based Data Center Networks", CoNEXT'13

[3] Mohammad Alizadeh, et al, "CONGA: Distributed Congestion-Aware Load Balancing for Datacenters", Sigcomm'14

[4] G. Dommety, "Key and Sequence Number Extensions to GRE", RFC 2890, September 2000

[5] D. Migault, S. Boutros, D. Wing, S. Krishnan,"Geneve Protocol Security Requirement", [I-D. draft-mglt-nvo3-geneve-security-requirements-03]




## Authors' Addresses


Haizhou Xiang
Huawei Technologies Co., Ltd.
Email: xianghaizhou@huawei.com

Yolanda Yu
Huawei Technologies Co., Ltd.
Email: yolanda.yu@huawei.com

Paul Congdon
Tallac Networks
paul.congdon@tallac.com

Jianglong Wang
China Telecom
Email: wangjl1.bri@chinatelecom.cn