

Network Virtualization Overlays Working
Group
Internet-Draft
Intended status: Standards Track
Expires: December 30, 2013

D. Wang
Q. Wu
Huawei
June 28, 2013

Proposed Control Plane requirements for Network Virtualization Overlays
[draft-wu-nvo3-nve2nve-06](#)

Abstract

This document looks at control plane aspect related to both tenant system to NVE control interface and NVE to Network Virtualization Authority (NVA) control interface NVE use to enable communication between tenant systems and focuses on NVE to NVA control interface, which is complementary to [[draft-kreeger-nvo3-hypervisor-nve-cp](#)] that describes the high level control plane requirements related to the interaction between tenant system and NVE when the two entities are not co-located on the same physical device.

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 December 30, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

Internet-Draft

NVE2NVA

June 2013

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Conventions Used in this Document	4
3.	NVO3 Control Plane Overview	5
4.	Tenant system information entry at the NVE and Network Virtualization Authority	6
4.1.	Tenant system information entry fields relationship . . .	7
4.2.	Forwarding functionality at the tenant system	8
5.	NVE to NVA Control Plane Protocol Functionality	11
5.1.	NVE to VN connect/disconnect notification	11
5.2.	Advertisement of Inner-outer address mapping associated with tenant system	11
5.3.	Tenant system information distribution	12
5.4.	VN context moving	13
6.	Hypervisor-to-NVE Control Plane Protocol Functionality	14
6.1.	Associate the NVE with VN context	14
6.2.	Localized forwarding at the same local NVE	14
7.	IANA Considerations	15
8.	Security Considerations	16
9.	References	17
9.1.	Normative References	17
9.2.	Informative References	17
Appendix A.	Change Log	18
A.1.	draft-wu-nvo3-nve2nve-06	18
A.2.	draft-wu-nvo3-nve2nve-05	18
A.3.	draft-wu-nvo3-nve2nve-04	18
	Authors' Addresses	20

1. Introduction

In [[I.D-ietf-nvo3-overlay-problem-statement](#)], two control planes are identified to realize an overlay solution:

- o NVE to Network Virtualization Authority (NVA) control plane.
- o Tenant system to NVE control plane.

Where NVE to NVA Control plane is used to deal with address mapping dissemination and Tenant System to NVE control plane is used to deal with VM attachment and detachment.

In [[I.D-ietf-nvo3-framework](#)], three control plane components are defined to build these two control planes and provide the following capabilities:

- o Auto-provision/service discovery
- o Address advertisement
- o Tunnel management

In [[I.D-fw-nvo3-server2vcenter](#)], the control interface between NVE and the Oracle backend system or Network Virtualization Authority (NVA) is defined to provide the following capabilities:

- o Enforce the network policy for each VM in the path from the NVE Edge associated with VM to the Tenant End System.
- o Populate forwarding table in the path from the NVE Edge associated with VM to the Tenant End System in the data center.
- o Populate mapping table in each NVE Edge that is in the virtual network across data centers under the control of the Director.

This document focuses on control plane aspect related to both tenant system to NVE control interface and NVE to Oracle control interface NVE use to enable communication between tenant systems, which is complementary to [[draft-kreeger-nvo3-hypervisor-nve-cp](#)] that describes the high level control plane requirements related to the interaction between tenant system and NVE when the two entities are not co-located on the same physical device.

[2.](#) Conventions Used in this Document

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

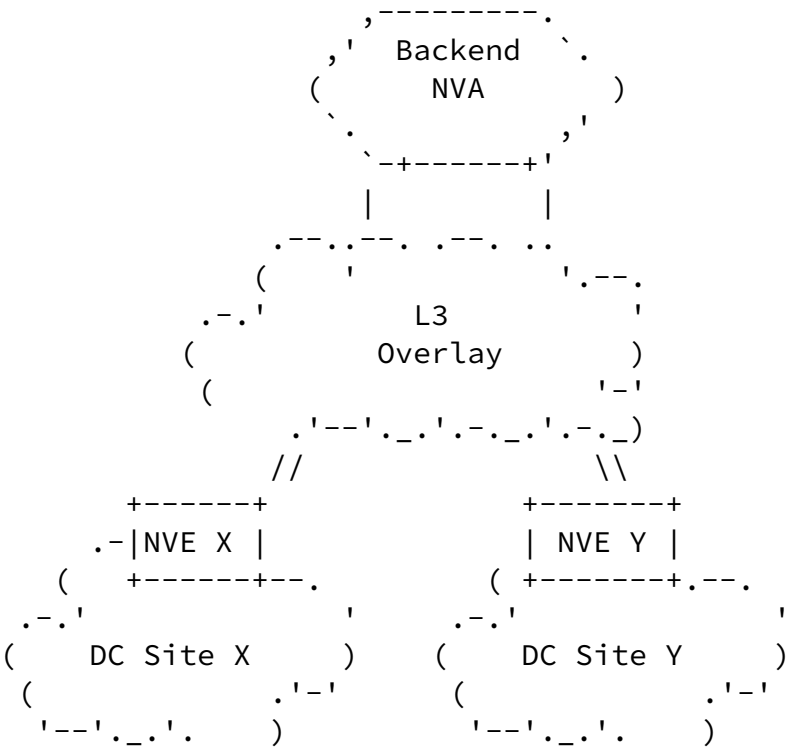
Tenant System:

A physical or virtual system that can play the role of a host, or a forwarding element such as a router, switch, firewall, etc. It belongs to a single tenant and connects to one or more VNs of that tenant.

vNIC:

A vNIC is similar to a physical NIC. Each virtual machine has one or more vNIC adapters that it uses to communicate with both the virtual and physical networks. Each vNIC has its own MAC address and can be assigned one or more IP addresses just like a NIC found in a non virtualized machine.

3. NV03 Control Plane Overview



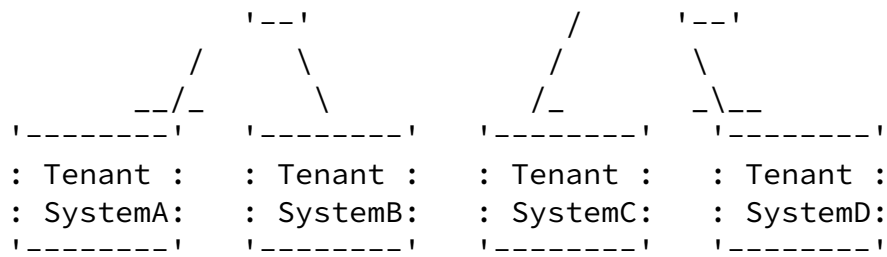


Figure 1: Example NV03 control plane Overview

4. Tenant system information entry at the NVE and Network Virtualization Authority

Every NVE pair(local NVE and remote NVE) associated with the tenant system MUST maintain at least one mapping table entry for each currently attached tenant system (In the case where TS has multiple tenant system interfaces, there may have multiple mapping table entry corresponding to one TS). In addition, the Network Virtualization Authority may also maintain a mapping table entry for each currently attached tenant system or each newly joined NVE. Each mapping table entry corresponds to the Tenant system connection to each VN or one Tenant system interface and conceptually may contain all or a sub set of the following fields:

- o The tunnel interface identifier (tunnel-if-id) of the tunnel

between the remote NVE and the local NVE where the tenant system is currently attached. The tunnel interface identifier is acquired during the tunnel creation.

- o The MAC address of the attached TS. This MAC address is obtained from auto-discovery protocol between Tenant System and its local NVE.
- o The IP address of the attached TS. This IP address is obtained from auto-discovery protocol between Tenant System and its local NVE.
- o The logical interface identifier (e.g., VLAN ID, internal vSwitch Interface ID connected to a Tenant System) of the access link between the tenant system and the local NVE. This field is required to associate Tenant System with local NVE if local NVE is an external NVE to Tenant system. It is internal to the local NVE and is also used to associate the tunnel to the access link where the tenant system is attached.
- o The MAC address of the local/remote NVE associated with the tenant system.
- o The IP address of the local/remote NVE associated with the tenant system.
- o The Identifier of VN context (VNID). This Identifier is obtained from auto-discovery protocol between Tenant System and its local NVE.
- o Lifetime of NVEs to keep table entries when pushed to or pulled from NVAs. While the table entries are pushed to NVA, they should be given a relatively long lifetime. Otherwise, they should be

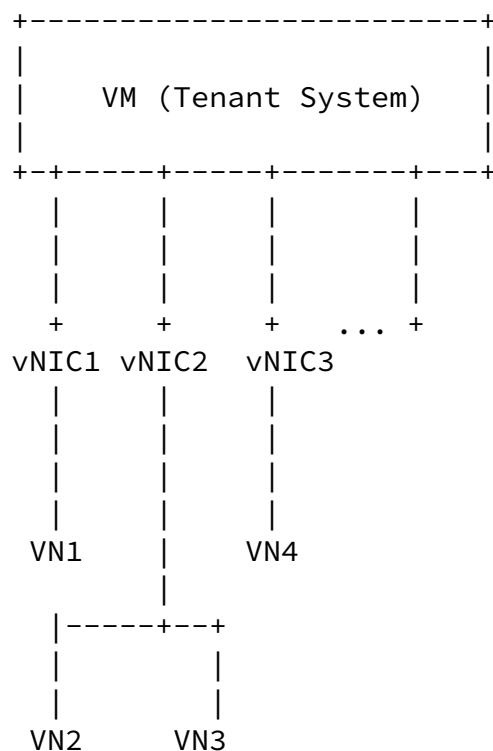
given a relatively short lifetime.

- o The operation code of tenant system. The operations includes shutdown, migration, startup, etc., which can be detected by the NVE on the access link between tenant system and the NVE.

[4.1.](#) Tenant system information entry fields relationship

One Tenant system is corresponding to one VM. Each Tenant System that is virtual system may have multiple vNIC adapters that it uses to communicate with both the virtual and physical networks. vNIC the tenant system has should belong to a single tenant. Each vNIC must be assigned with one unique MAC address. vNIC MAC address may be modified or assigned with a new MAC address at some time. When multiple vNICs hosted in the same VM connect to multiple VNs, it is allowed that some of these vNICs may connect to different VNs through the same NVE.

Each tenant system uses TSI to interface with VNI at the NVE via VAP. Each TSI can be identified by an identifier which the tenant system assigns to the TSI. Each VAP can be identified by the logical interface identifiers (e.g., VLAN ID, internal vSwitch Interface ID connected to a VM) which the NVE assigns to the VAP. In order to establish the network connection between tenant system and NVE and associate tenant system and NVE with the same VN, VNID should be used to correlate one TSI to one VAP that belong to the same VNI.



TSIa [VNID1,MAC addr1]corresponding to vNIC1

TSIb [VNID2,MAC addr2]corresponding to vNIC2

TSIc [VNID3,MAC addr3]corresponding to vNIC2

TSId [VNID4,MAC addr4]corresponding to vNIC3

Figure 2. Tenant System information Hierarchy

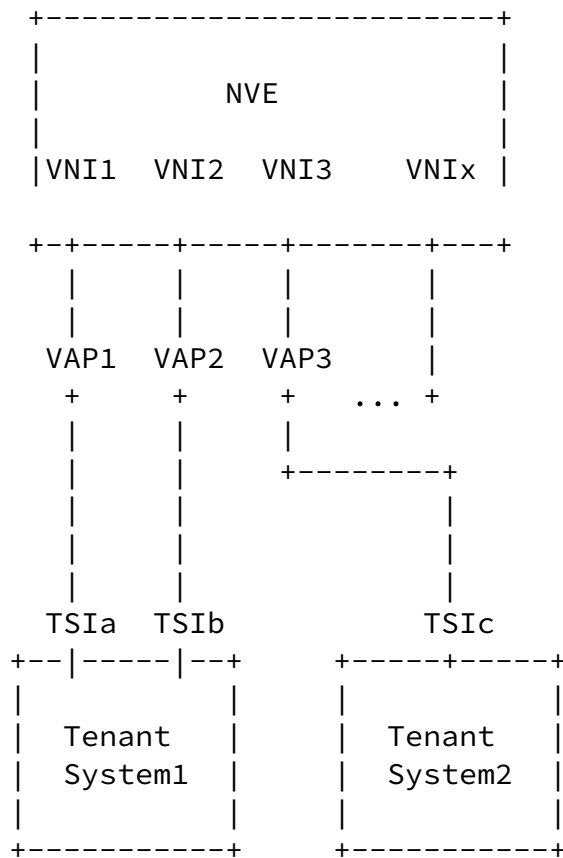


Figure 3. Interfaces between Tenant system and NVE

[4.2.](#) Forwarding functionality at the tenant system

When the tenant system A plays the role of forwarding functionality to connect two VNs, the following three cases should be considered.

- (a) Both two VNs support Layer 3 forwarding;
- (b) Both two VNs support layer 2 forwarding;

- (c) One VN supports Layer3 forwarding and the other VN supports layer 2 forwarding;

For (a), tenant system A or external system that is close to tenant system A should support layer 3 forwarding functionality. When source tenant system in one VN communicates with destination tenant system in another VN through the tenant system A, if tenant system A support layer 3 forwarding, the tenant system A should forward IP packets on the behalf of source Tenant System and destination tenant system irrespective of data plane encapsulation format(e.g., VXLAN or NVGREW, MPLS over GRE). If two VNs use different data plane encapsulation format, tenant system A should also support converting one data plane encapsulation format into another. If tenant system A doesn't support layer 3 forwarding, the external system that is close to tenant system A should associate TSI to local NVE using information like VNID, TS MAC address and VLAN tag information and forward IP packets on the behalf of source tenant system and destination tenant system.

For (b), tenant system A vNIC or external system that is close to tenant system A should support layer 2 forwarding functionality. When source tenant system in one VN communicates with destination tenant system in another VN through the tenant system A, if tenant system A support layer 2 forwarding, the tenant system A should know which tenant systems connecting to itself are allowed for layer 2 forwarding and then forward layer 2 frames on the behalf of source Tenant System and destination tenant system based on forwarding allowed list. If two layer 2 VNs support different data plane encapsulation format, the tenant system A should also support converting one data plane encapsulation format to another. If tenant system A doesn't support layer 2 forwarding, the external system that is close to tenant system A should associate TSI to local NVE using information like VNID, TS MAC address and VLAN tag information and forward layer 2 frames on the behalf of source tenant system and destination tenant system.

For (c), tenant system A or external system that is close to tenant system A should support both layer 2 forwarding and layer 3 forwarding. When source tenant systems in layer 2 VN communicates with destination tenant system in layer 3 VN through the tenant system A, if tenant system A support both layer 2 and layer 3 forwarding the tenant system A should support translating layer 2 frame into layer 3 packet and forward traffic between layer 2 VN and layer 3 VN. If two VNs support different data plane encapsulation format, the tenant system A should also support converting one data plane encapsulation format to another. If tenant system A doesn't

support layer 2 forwarding or layer3 forwarding, the external system that is close to tenant system A should associate TSI to local NVE

using information like VNID,TS MAC address and VLAN tag information and forward traffic on the behalf of source tenant system and destination tenant system.

When the tenant system A plays the role of interconnection functionality to connect between VN and Non-VN, suppose source tenant system in one VN communicates with destination end device in Non-VN environment through the tenant system A, the tenant system A acts as NV03 GW between VN and Non-VN in this case peering with other Gateways and should be explicitly configured with a list of destination MAC addresses that allows passed to the Non-VN environment and perform translation between VNID and Non-VN label when forwarding traffic between VN interface and Non-VN interface. For outgoing frames on VN connected interface, the tenant system A decapsulates NV03 outer header and forwards the inner frame to Non-VN environment based on configured allowed list. For incoming frames on non-VN connected interface(e.g.,WAN interface), the tenant system A should map the incoming frames from end device to specific VN based on inner Ethernet frame information (e.g., VLAN ID). The mapping table is setup at the tenant system A to perform VNID lookup in VN and label lookup in the Non-VN environment.

[5.](#) NVE to NVA Control Plane Protocol Functionality

The core functional entities for NVE to NVA Control plane infrastructure are the NVE and Network Virtualization Authority. The Network Virtualization Authority is responsible for maintaining the tenant system reachability state and is the topological anchor point for the Tenant system information (e.g., Tenant system MAC address, IP address, VN Name, VNID, local NVE addresses, remote NVE addresses). There can be multiple NVAs in a VN each serving a different group of tenant system. The NVE is the entity that performs the inner-outer address mapping and VN Name to VNID mapping management on the request of NVE, and it resides on the NVE or an external network device separately from NVE. The NVE is responsible for detecting the tenant system operations (e.g., Shutdown, Migration, Startup) on the access link between tenant system and NVE and for advertise VN context information associated with tenant system to NVA.

[5.1.](#) NVE to VN connect/disconnect notification

When a tenant system connects to one VN by attaching to a local NVE, The tenant system should also inform the attached local NVE which VN context the tenant system belong to. the local NVE should also be added into VN context together with tenant system information and report VN membership to Network Virtualization Authority (NVA). This helps Network Virtualization Authority know to which NVE a group of the tenant systems are attached or current location of these tenant systems. When the last tenant system is disconnected to one VN through one local NVE, this local NVE should also be removed from VN context. This should also be updated to Network Virtualization Authority and let Network Virtualization Authority know that there are no tenant system associated with that NVE.

5.2. Advertisement of Inner-outer address mapping associated with tenant system

In order to enable tenant system A to communicate with any tenant system that is not under the same local NVE, the inner-outer address mapping (Destination Tenant system L2/L3 address to remote NVE (i.e., egress tunnel endpoint)) that maps a final destination address to the proper tunnel should be distributed to all the remote NVEs that belong to the same VN even though there is no tenant system which communicates with tenant system A behind that remote NVE. Alternatively, the inner-outer address mapping can be distributed to the NVA and then NVA advertise inner-outer address mapping to the corresponding remote NVE according to VN membership established using NVE connect/disconnect to VN notification. When NVA is embedded within NVE, there is no need for a standardized protocol between the NVE and NVA, as the interaction is implemented via software on a

single device. For inner-outer address mapping distribution to NVE, in one approach, BGP protocol can be used to advertise such mapping information, alternatively, each NVE may push inner-outer mapping to the Network Virtualization Authority using NVE connect/disconnect to VN notification or other NVE-to-NVA protocol.

5.3. Tenant system information distribution

Data plane learning can be used to build mapping table without need for control plane protocol. However it requires each data packet to be flooded to the whole VN. In order to eliminate flooding introduced by data plane learning, a control protocol is needed to provide inner-outer address mapping and other information associated with tenant system from NVA to the corresponding NVE. For tenant system information distribution, one approach is the tenant system information is pushed by NVA to the NVE. If the destination packet over the logical tunnel arriving at the NVE can't be found in its inner-outer mapping table that are pushed down from the NVA, the NVA could be configured to simply drop the data frame, or flood it to all other remote NVE that belong to the same VN if NVE knows VN membership. If an NVE lost its connectivity to its NVA, it MUST ignore any Pushed data from the NVA because the pushed data may be outdated or not valid. When there might have multiple NVA holding the mapping information for the tenant systems in the VN and push the same tenant system information to the same NVE (i.e., conflict

occurs), the destination packet can be tagged with different priority, higher priority data take precedence.

Alternatively, NVE can send pull request to NVA for the tenant system information. Each Pull request can have multiple queries for different Tenant Systems. The pull request can be triggered by An edge node (NVE) receives an ingress data frame with a destination whose attached edge (NVE) is unknown, or the edge node (NVE) receives an ingress ARP/ND request for a target whose link address (MAC) or attached edge (NVE) is unknown. If the NVA can be configured to prohibit some NVEs to get tenant system information from NVA or the NVA may not have entry matching tenant system information asked by NVE. Then NVA can indicate in the pull response that the target being queried is not available or NVE is not allowed to access information, otherwise, the NVE should return the valid inner-outer address mapping with the valid timer indicating how long the entry can be cached by the edge (NVE). While waiting for query response from NVA, the NVE has to buffer the subsequent data packets with destination address to the same target. The buffer could overflow before the NVE gets the response from NVA. If no response is received to a NVA within a configurable timeout, the request should be re-transmitted up to a configurable number of times. When NVE caching entry pulled from NVA is expired, both NVE and NVA should

remove invalid NVE caching entry. When one tenant system is detached from one NVE and move to another, the inner-out address mapping established in the pervious NVE is not valid. In such cases, the inner-outer address mapping should be removed from the previous NVE and the new inner-outer mapping should be created at the new NVE to which the tenant system is currently attached. Such inner-outer mapping should be updated at NVA. If an NVE lost its connectivity to its NVA, the cached entry should be removed from the NVE to which the tenant system is currently attached.

[5.4.](#) VN context moving

In some cases, one tenant system may be detached from one NVE and move to another NVE. In such cases, the VN context should be moved from the NVE to which the tenant system was previously attached to the new NVE to which the tenant system is currently attached. In order to achieve this, the per tenant system VN context including VN profile can be maintained at the Network Virtualization Authority and

be retrieved at the new place based on the VN Identifier (VNID). 6.
Hypervisor-to-NVE Control Plane Protocol Functionality

[6.](#) Hypervisor-to-NVE Control Plane Protocol Functionality

[6.1.](#) Associate the NVE with VN context

The VN context includes a set of configuration attributes defining access and tunnel policies and (L2 and/or L3) forwarding functions. When a Tenant System is attached to a local NVE, a VN network instance should be allocated to the local NVE. The tenant system should be associated with the specific VN context using virtual Network Instance(VNI). The tenant system should also inform the attached local NVE which VN context the tenant system belong to.

Therefore the VN context can be bound with the data path from the tenant system to the local NVE and the tunnel from local NVE associated with the tenant system and all the remote NVEs that belong to the same VN as the local NVE. For the data path from the tenant system and the local NVE, the network policy can be installed on the underlying switched network and forwarding tables also can be populated to each network elements in the underlying network based on the specific VNI associated with the tenant system. For the tunnel from local NVE to the remote NVEs, the traffic engineering information can be applied to each tunnel based on VNI associated with the tenant system.

[6.2.](#) Localized forwarding at the same local NVE

In some cases, two tenant systems may be attached to the same local NVE. In order to allow the NVE to locally forward traffic between two tenant systems that are attached to the same NVE, the inner-outer address mapping that maps a final destination address to the proper tunnel should be populated at the local NVE.

In some cases, two tenant systems may connect to the different VNs through the same interconnection functionality (Data Center Gateway), in order to allow two tenant systems communication between two VNs, the mapping table that maps a final destination address to the proper tunnel should also be populated in both NVE associated with two communicated tenant system and the interconnection functionality associated corresponding NVE. In this case, the interconnection functionality may trigger both NVE associated with two tenant system to establish tunnel directly and allow traffic between these two tenant systems bypass itself.

[7.](#) IANA Considerations

This document has no actions for IANA.

[8.](#) Security Considerations

TBC.

Internet-Draft

NVE2NVA

June 2013

[9.](#) References

[9.1.](#) Normative References

[I.D-ietf-nvo3-framework]

Lasserre, M., "Framework for DC Network Virtualization",
ID [draft-ietf-nvo3-framework-00](#), September 2012.

[I.D-ietf-nvo3-overlay-problem-statement]

Narten, T., "Problem Statement: Overlays for Network
Virtualization",
ID [draft-ietf-nvo3-overlay-problem-statement-02](#),
February 2013.

[I.D-kreeger-nvo3-hypervisor-nve-cp]

Kreeger, L., "Network Virtualization Hypervisor-to-NVE
Overlay Control Protocol Requirements",
ID [draft-kreeger-nvo3-hypervisor-nve-cp-01](#), February 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", March 1997.

[9.2.](#) Informative References

[I.D-fw-nvo3-server2vcenter]

Wu, Q. and R. Scott, "Network Virtualization
Architecture", ID [draft-fw-nvo3-server2vcenter-01](#),
January 2013.

[Appendix A](#). Change Log

Note to the RFC-Editor: please remove this section prior to publication as an RFC.

[A.1](#). [draft-wu-nvo3-nve2nve-06](#)

The following are the major changes to previous version :

- o Remove [section 7](#).
- o Add new sub[section 5.1](#) to discuss NVE to VN connect/disconnect notification.
- o Add new sub[section 5.2](#) to discuss Inner-outer address mapping associated with tenant system advertisement.
- o Add new sub[section 5.3](#) to discuss Tenant system information distribution.
- o Add new sub[section 6.1](#) to discuss Associate the NVE with VN context.
- o Add new sub[section 6.1](#) to discuss localized forwarding at the same local NVE.

[A.2](#). [draft-wu-nvo3-nve2nve-05](#)

The following are the major changes to previous version :

- o Remove distinction between pNIC and vNIC and restrict tenant system to the one that is virtual system and has vNICs

- o Add one new figure and Using VAP and TSI to establish association between tenant system and NVE that belong to the same VN.
- o Delete Oracle Backend System term.
- o Replace interconnection functionality with forwarding functionality.

[A.3. draft-wu-nvo3-nve2nve-04](#)

The following are the major changes to previous version :

- o Rewording some vNICs in the document with TSI.

Wang & Wu

Expires December 30, 2013

[Page 18]

Internet-Draft

NVE2NVA

June 2013

- o Clarify the relation between VM,Tenant System,TSI and distinguish network, network elements from identifier for network or network elements.
- o Distinguish pNIC from vNIC.
- o Using TSI Identifier to identify each TSI
- o Support multiple TSI for multiple simultaneous connection and using BID to distinguish different TSI belong to the same vNIC.

Authors' Addresses

Danhua Wang
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: wangdanhua@huawei.com

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

