Internet Engineering Task Force
Internet Draft
Intended status: Informational
Expires: September 2012

Marc Lasserre
Florin Balus
Alcatel-Lucent

Thomas Morin
France Telecom Orange

March 5, 2012

Framework for DC Network Virtualization
draft-lasserre-nvo3-framework-00.txt

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), 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/ietf/1id-abstracts.txt

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

This Internet-Draft will expire on September 5, 2012.

Copyright Notice

carefully, as they describe your rights and restrictions with respect to this document.

Abstract

Several IETF drafts relate to the use of overlay networks to support large scale virtual data centers. This draft provides a framework for Network Virtualization over L3 (NVO3) and is intended to help plan a set of work items in order to provide a complete solution set. It defines a logical view of the main components with the intention of streamlining the terminology and focusing the solution set.

Table of Contents

1. Introduction

This document provides a framework for Data Center Network Virtualization over L3 tunnels. This framework is intended to aid in standardizing protocols and mechanisms to support large scale network virtualization for data centers.

Several IETF drafts relate to the use of overlay networks for data centers.

[NVOPS] defines the rationale for using overlay networks in order to build large data center networks. The use of virtualization leads to a very large number of communication domains and end systems to cope with. Existing virtual network models used for data center networks have known limitations, specifically in the context of multiple tenants, that have also been described in various sections of [VXLAN], [NVGRE], and [DCVPN]. These issues can be summarized as:

- Limited VLAN space
- FIB explosion due to handling of large number of MACs/IP addresses
- Spanning Tree limitations
- Excessive ARP handling
- Broadcast storms
- Inefficient Broadcast/Multicast handling
- Limited mobility/portability support
- Lack of service auto-discovery

[VXLAN], [NVGRE], [STT] and [DCVPN] describe the use of overlay techniques that address some of these issues.

[OVCPREQ] describes the requirements for a control plane protocol required by overlay border nodes to exchange overlay mappings.

This document provides reference models and functional components of data center overlay networks as well as a discussion of technical issues that have to be addressed in the design of standards and mechanisms for large scale data centers.

1.1. 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 RFC-2119 [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

1.2. General terminology

Some general terminology is defined here. Terminology specific to this memo is introduced as needed in later sections.

DC: Data Center

ELAN: MEF ELAN, multipoint to multipoint Ethernet service

EVPN: Ethernet VPN as defined in [EVPN]

1.3. DC network architecture

A generic architecture for Data Centers is depicted in Figure 1:

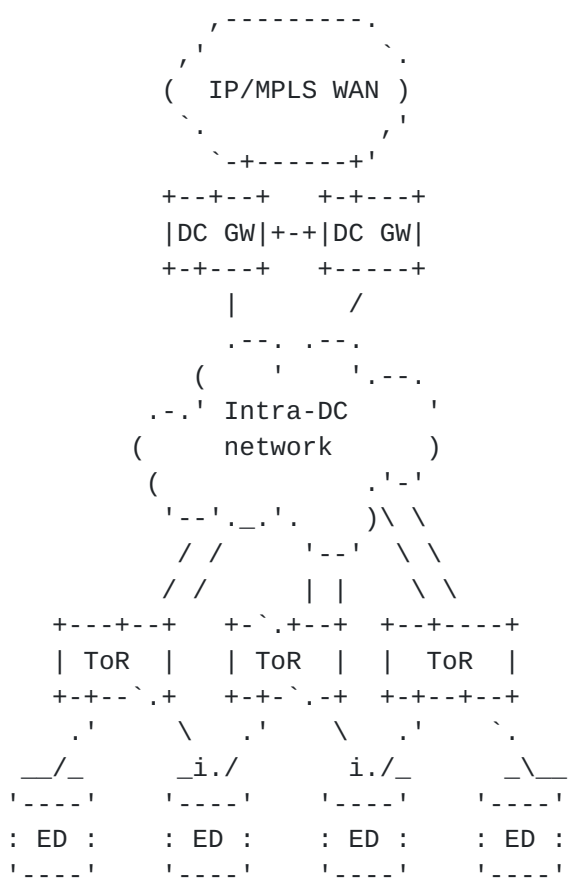

Figure 1 : A Generic Architecture for Data Centers

An example of multi-tier DC network architecture is presented in this figure. A cloud network is composed of intra-Data Center (DC) networks and network services, and, inter-DC network and network connectivity services. Depending upon the scale, DC distribution, operations model, Capex and Opex aspects, DC networking elements can act as strict L2 switches and/or provide IP routing capabilities, including also service virtualization.

In some DC architectures, it is possible that some tier layers provide L2 and/or L3 services, are collapsed, and that Internet connectivity, inter-DC connectivity and VPN support are handled by a smaller number of nodes. Nevertheless, one can assume that the functional blocks fit with the architecture above.

The following components can be present in a DC:

- o End Device (ED): a DC resource to which the networking service is provided. ED may be a compute resource (server or server

blade), storage component or a network appliance (firewall, load-balancer, IPsec gateway). Alternatively, the End Device may include software based networking functions used to interconnect multiple IP hosts. An example of soft networking is the virtual switch in the server blades, used to interconnect multiple virtual machines (VMs). ED may be single or multi-homed to the Top of Rack switches (ToRs).

- o Top of Rack (ToR): Hardware-based Ethernet switch aggregating all Ethernet links from the End Devices in a rack representing the entry point in the physical DC network for the hosts. ToRs may also provide routing functionality, virtual IP network connectivity, or Layer2 tunneling over IP for instance. ToRs are usually multi-homed to switches in the Intra-DC network. Other deployment scenarios may use an EoR (End of Row) switch to provide similar function as a ToR.

- o Intra-DC Network: High capacity network composed of core switches aggregating multiple ToRs. Core switches are usually Ethernet switches but can also support routing capabilities.

- o DC GW: Gateway to the outside world providing DC Interconnect and connectivity to Internet and VPN customers. In the current DC network model, this may be simply a Router connected to the Internet and/or an IPVPN/L2VPN PE. Some network implementations may dedicate DC GWs for different connectivity types (e.g., a DC GW for Internet, and another for VPN).

We use throughout this document the term "End System" to define an end system of a particular tenant, which can be for instance a virtual machine (VM), a non-virtualized server, or a non-virtualized network appliance. One or more End Systems can be part of an ED.

1.4. Tenant networking view

The DC network architecture is used to provide L2 and/or L3 service connectivity to each tenant. An example is depicted in Figure 2:

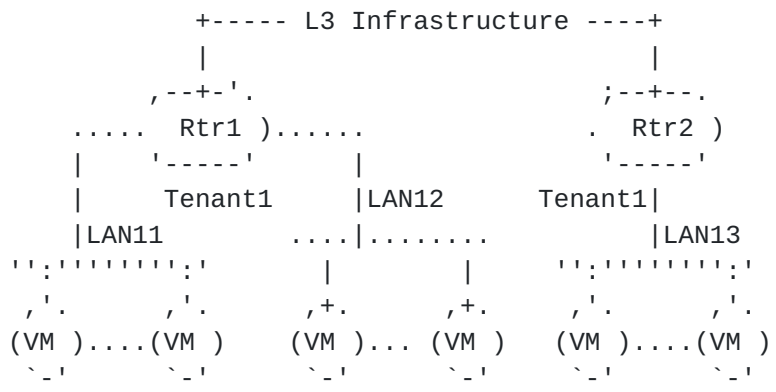

Figure 2 : Logical Service connectivity for a single tenant

In this example one or more L3 contexts and one or more LANs (e.g., one per Application) running on DC switches are assigned for DC tenant 1.

For a multi-tenant DC, a virtualized version of this type of service connectivity needs to be provided for each tenant by the Network Virtualization solution.

2. Reference Models

2.1. Generic Reference Model

The following diagram shows a DC reference model for network virtualization using Layer3 overlays where edge devices provide a logical interconnect between end systems that belong to specific tenant network.

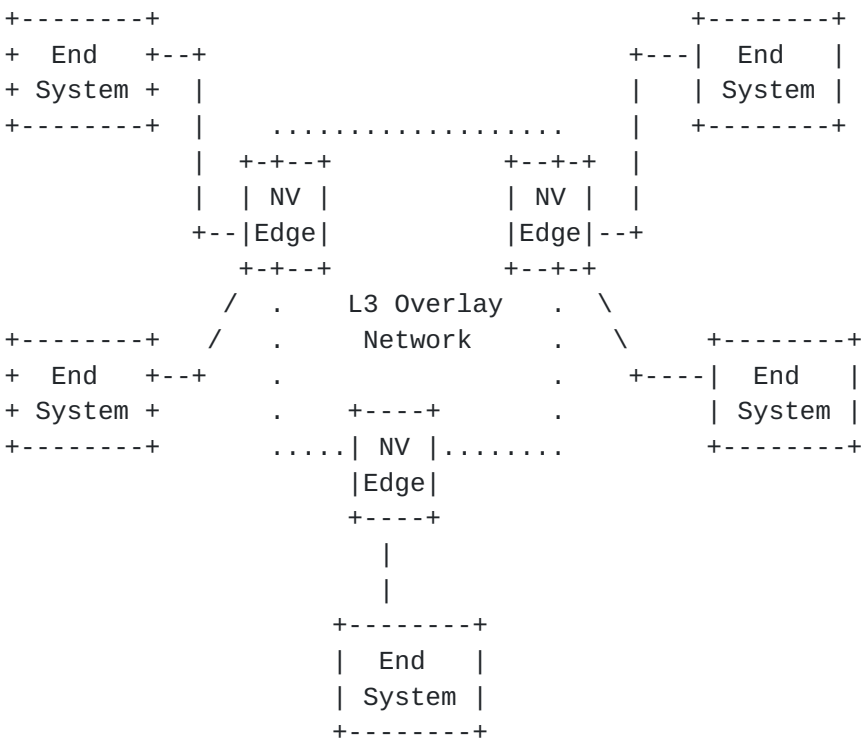

Figure 3 : Generic reference model for DC network virtualization over a Layer3 infrastructure

An End System attaches to a Network Virtualization Edge (NVE) node, either directly or via a switched network (typically Ethernet). Examples of DC End Systems are host machines, including Virtual Machines, Network Appliances or Storage Systems.

The NVE implements network virtualization functions that allow for L2 and/or L3 tenant separation and for hiding tenant addressing information (MAC and IP addresses), tenant-related control plane activity and service contexts from the Routed Backbone nodes.

Core nodes utilize L3 techniques to interconnect NVE nodes in support of the overlay network. These devices perform forwarding based on outer L3 tunnel header, and generally do not maintain per tenant-service state albeit some applications (e.g., multicast) may require control plane or forwarding plane information that pertain to a tenant, group of tenants, tenant service or a set of services that belong to one or more tunnels. When such tenant or tenant-

service related information is maintained in the core, overlay virtualization provides knobs to control the magnitude of that information.

2.2. NVE Reference Model

The NVE is composed of a tenant service instance that end systems interface with and an overlay module that provides tunneling overlay functions (e.g. encapsulation/decapsulation of tenant traffic from/to the tenant forwarding instance, tenant identification and mapping, etc), as described in figure 4:

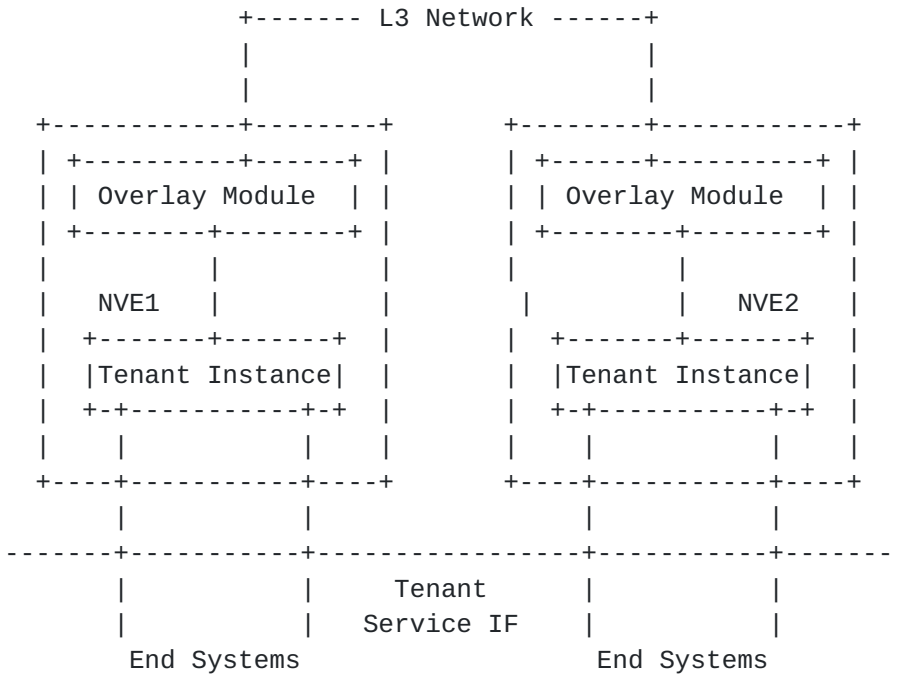

Figure 4 : Generic reference model for NV Edge

Note that some NVE functions (e.g. data plane and control plane functions) may reside in one device or they may be distributed between multiple devices.

2.3. NVE Service Types

NVE components may be used to provide different types of virtualized service connectivity. This section defines the service types and associated attributes

2.3.1. L2 NVE providing Ethernet LAN-like service

L2 NVE implements Ethernet LAN emulation (ELAN), an Ethernet based multipoint service where the End Systems appear to be interconnected by a LAN environment over a set of L3 tunnels. It provides per tenant virtual switching instance with MAC addressing isolation and L3 tunnel encapsulation across the core.

2.3.2. L3 NVE providing IP/VRF-like service

Virtualized IP routing and forwarding is similar from a service definition perspective with IETF IP VPN (e.g., BGP/MPLS IPVPN and IPsec VPNs). It provides per tenant routing instance with addressing isolation and L3 tunnel encapsulation across the core.

3. Functional components

This section breaks down the Network Virtualization architecture into functional components to make it easier to discuss solution options for different modules.

This version of the document gives an overview of generic functional components that are shared between L2 and L3 service types. Details specific for each service type will be added in future revisions.

3.1. Generic service virtualization components

A Network Virtualization solution is built around a number of functional components as depicted in Figure 5:

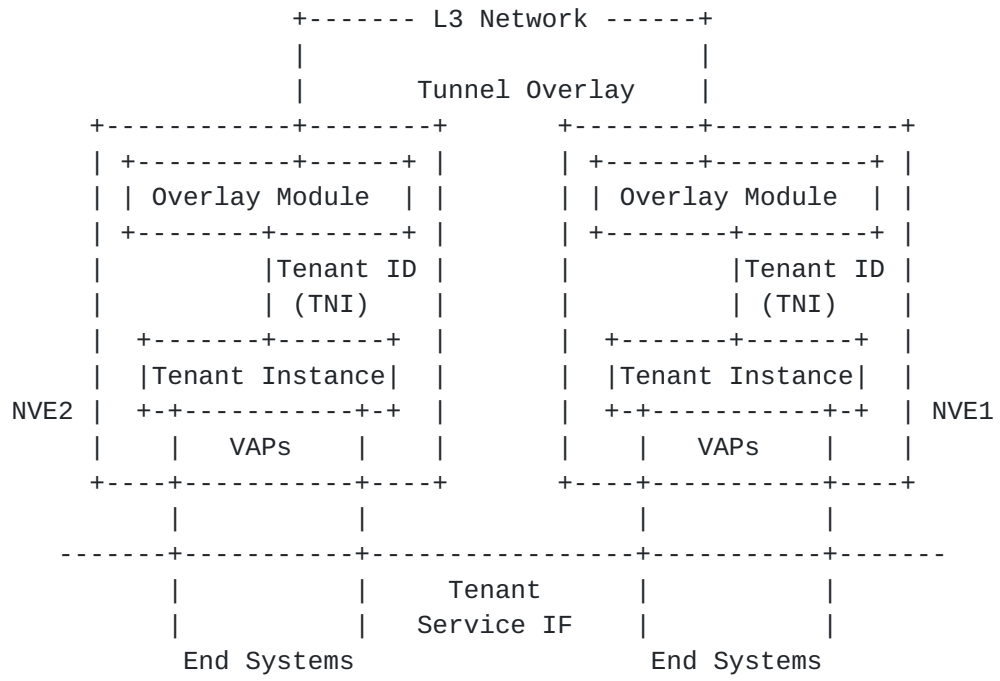

Figure 5 : Generic reference model for NV Edge

3.1.1. Virtual Access Points (VAPs)

End Systems are connected to the Tenant Instance through Virtual Access Points (VAPs). The VAPs can be in reality physical ports on a ToR or virtual ports identified through logical interface identifiers (VLANs, internal VSwitch Interface ID leading to a VM).

3.1.2. Tenant Instance

The Tenant Instance represents a set of configuration attributes defining access and tunnel policies and (L2 and/or L3) forwarding functions.

Per tenant FIB tables and control plane protocol instances are used to maintain separate private contexts between tenants. Hence tenants are free to use their own addressing schemes without concerns about address overlapping with other tenants.

3.1.3. Overlay Modules and Tenant ID

The Overlay module provides tunneling overlay functions: tunnel initiation/termination, encapsulation/decapsulation of frames from VAPs/L3 Backbone and may provide for transit forwarding of IP traffic (e.g., transparent tunnel forwarding).

In a multi-tenant context, the tunnel aggregates frames from/to different Tenant Instances. Tenant identification and traffic demultiplexing are based on the Tenant Identifier (TNI).

At least two possible approaches for TNI should be considered:

- One ID per Tenant: A globally unique (on a per-DC administrative domain) Tenant ID is used to identify the related Tenant instances. An example of this approach is the use of IEEE VLAN or ISID tags to provide virtual L2 domains.

- One ID per Tenant Instance (TNI): A per-tenant local ID is automatically generated by the egress NVE and usually distributed by a control plane protocol to all the related NVEs. An example of this approach is the use of per VRF MPLS labels in IP VPN [RFC4364].

- One ID per VAP: A per-VAP local ID is assigned and usually distributed by a control plane protocol. An example of this approach is the use of per CE-PE MPLS labels in IP VPN [RFC4364].

Note that when using one ID per TNI or VAP, an additional global identifier may be used by the control plane to identify the Tenant context.

3.1.4. Tunnel Overlays and Encapsulation options

Once the TNI is added to the frame an IP Tunnel encapsulation is used to transport the frame to the destination NVE. The backbone devices do not usually keep any per service state, simply forwarding the frames based on the outer tunnel header.

Different IP tunneling options (GRE/L2TP/IPSec) are already available for both Ethernet and IP formats. A UDP/IP option is described in [VXLAN].

3.1.5. Use of Control Plane Protocols

A set of control plane components may be used to provide certain functions related to auto-provisioning, route advertisement, efficient BUM handling, or ARP reduction for instance, as discussed in section 4.2.

Further details will be provided in a subsequent revision of this document.

3.2. Service Overlay Topologies

A number of service topologies may be used to optimize the service connectivity and to address NVE performance limitations.

The topology described in Figure 3 suggests the use of a tunnel mesh between the NVEs where each tenant instance is one hop away from a service processing perspective. Partial mesh topologies and an NVE hierarchy may be used where certain NVEs may act as service transit points.

4. Key aspects of overlay networks

4.1. Pros & Cons

An overlay network is a layer of virtual network topology on top of the physical network.

Overlay networks offer the following key advantages:

- Tunnel state management is handled at the edge of the network. Intermediate transport nodes are unaware of such state, provided that flood containment or multicast capabilities on a per-tenant basis are not required from the core network

- Tunnels are used to aggregate traffic and hence offer the advantage of minimizing the amount of forwarding state required within the underlay network

- Decoupling of the overlay addresses (MAC and IP) used by VMs from the underlay network. This offers a clear separation between addresses used within the overlay and the underlay networks and it enables the use of overlapping addresses spaces by end systems

- Support of a large number of virtual network identifiers

Overlay networks also create several challenges:

- Overlay networks have no controls of underlay networks and lack critical network information
    - Overlays typically probe the network to measure link properties, such as available bandwidth or packet loss rate. It is difficult to accurately evaluate network properties. It might be preferable for the underlay network to expose usage and performance information.
- Miscommunication between overlay and underlay networks can lead to an inefficient usage of network resources.
- Fairness of resource sharing and collaboration among end-nodes in overlay networks are two critical issues
- When multiple overlays co-exist on top of a common underlay network, the lack of communication between overlays can lead to performance issues.
- Overlaid traffic may not traverse firewalls and NAT devices.
- Multicast service scalability. Multicast support may be required in the overlay network to address for each tenant flood containment or efficient multicast handling.

4.2. Overlay issues to consider

4.2.1. End System to Overlay Network Mapping

NVEs must be able to select the appropriate Tenant Instance for each End System. This is based on state information that is often distributed from external entities. For example, in a VM environment, this information is provided by compute management systems, since these are the only entities that have visibility on which VM belongs to which tenant.

A standard mechanism for communicating this information between End Systems and the network is required. Note, that depending on the implementation this control interface can be between compute management and a virtual switch or between compute management and/or End Systems and a ToR switch.

In either case the protocol must provide appropriate security and authentication mechanisms to verify that End System information is not spoofed or altered. This is one of the most critical aspects for providing integrity and tenant isolation in the system.

4.2.2. Address to tunnel mapping

As traffic reaches an ingress NVE, a lookup is performed to determine which tunnel the packet needs to be sent to. It is then encapsulated with a tunnel header containing the destination address of the egress overlay node. Intermediate nodes (between the ingress and egress NVEs) switch or route traffic based upon the outer destination address.

One key step in this process consists of mapping a final destination address to the proper tunnel. NVEs are responsible for maintaining such mappings in their lookup tables.

Several ways of populating these lookup tables are possible: data plane driven, control plane driven or management plane driven. Destination addresses can be dynamically learned as would occur in standard bridges, or they can be populated by a control plane protocol or a network management system.

4.2.3. Data plane vs Control plane driven

Dynamic (data plane) learning implies that flooding of unknown destinations be supported and hence implies that broadcast and/or multicast be supported. Multicasting in the core network for dynamic learning can lead to significant scalability limitations. Specific forwarding rules must be enforced to prevent loops from happening. This can be achieved using a spanning tree protocol or a shortest path tree, or using a split-horizon mesh.

A control plane protocol can distribute this information instead. As an example, [EVPN] describes a procedure to distribute the VM MACs and build forwarding entries in each Tenant Instance. Alternative control plane protocols and/or options are applicable.

It should be noted that the amount of state to be distributed is a function of the number of virtual machines. Different forms of caching can also be utilized to minimize state distribution between the various elements.

4.2.4. Coordination between data plane and control plane

Often a combination of data plane and control based learning is necessary. Learning is applied towards end-user facing ports whereas distribution is used on the tunnel ports. Coordination between the learning engine and the control protocol is needed such that when a new address gets learned or an old address is removed, it triggers the local control plane to distribute this information to its peers.

4.2.5. Handling Broadcast, Unknown Unicast and Multicast (BUM) traffic

There are two techniques to support packet replication needed for broadcast, unknown unicast and multicast:

- Ingress replication
- Use of core multicast trees

There is a bandwidth vs state trade-off between the two approaches. Depending upon the degree of replication required (i.e. the number of hosts per group) and the amount of multicast state to maintain, trading bandwidth for state is of consideration.

When the number of hosts per group is large, the use of core multicast trees may be more appropriate. When the number of hosts is small (e.g. 2-3), ingress replication may not be an issue.

Depending upon the size of the data center network and hence the number of (S,G) entries, but also the duration of multicast flows, the use of core multicast trees can be a challenge.

When flows are well known, it is possible to pre-provision such multicast trees. However, it is often difficult to predict application flows ahead of time, and hence programming of (S,G) entries for short-lived flows could be impractical.

A possible trade-off is to use in the core shared multicast trees as opposed to dedicated multicast trees.

4.2.6. Path MTU

When using overlay tunneling, an outer header is added to the original frame. This can cause the MTU of the path to the egress tunnel endpoint to be exceeded.

In this section, we will only consider the case of an IP overlay.

It is usually not desirable to rely on IP fragmentation for performance reasons. Ideally, the interface MTU as seen by an end system is adjusted such that no fragmentation is needed. TCP will adjust its maximum segment size accordingly.

It is possible for the MTU to be configured manually or to be discovered dynamically. Various Path MTU discovery techniques exist in order to determine the proper MTU size to use:

- Classical ICMP-based MTU Path Discovery [RFC1191] [RFC1981]
    - End systems rely on ICMP messages to discover the MTU of the end-to-end path to its destination. This method is not always possible, such as when traversing middle boxes (e.g. firewalls) which disable ICMP for security reasons
- Extended MTU Path Discovery techniques such as defined in [RFC4821]

It is also possible to rely on the overlay layer to perform segmentation and reassembly operations without relying on the end systems to know about the end-to-end MTU. The assumption is that some hardware assist is available on the NVE node to perform such SAR operations. Such a mechanism is described in [STT]. However, fragmentation by the overlay layer can lead to performance and congestion issues due to TCP dynamics and might require new congestion avoidance mechanisms from then underlay network [FLOYD].

Finally, the underlay network may be designed in such a way that the MTU can accommodate the extra tunnel overhead.

4.2.7. NVE location trade-offs

In the case of DC traffic, traffic originated from a VM is native Ethernet traffic. This traffic can be switched by a local VM switch or ToR switch and then by a DC gateway. The NVE function can be embedded within any of these elements.

There are several criteria to consider when deciding where the NVE processing boundary happens:

- Processing and memory requirements
    - Datapath (e.g. lookups, filtering, encapsulation/decapsulation)

- Control plane processing (e.g. routing, signaling, OAM)

- FIB/RIB size
- Multicast support
  - Routing protocols
  - Packet replication capability
- Fragmentation support
- QoS transparency
- Resiliency

4.2.8. Interaction between network overlays and underlays

When multiple overlays co-exist on top of a common underlay network, this can cause some performance issues. These overlays have partially overlapping paths and nodes.

Each overlay is selfish by nature in that it sends traffic so as to optimize its own performance without considering the impact on other overlays, unless the underlay tunnels are traffic engineered on a per overlay basis so as to avoid sharing underlay resources.

Better visibility between overlays and underlays can be achieved by providing mechanisms to exchange information about:

- Performance metrics (throughput, delay, loss, jitter)
- Cost metrics

5. Security Considerations

The tenant to overlay mapping function can introduce significant security risks if appropriate protocols are not used that can support mutual authentication.

No other new security issues are introduced beyond those described already in the related L2VPN and L3VPN RFCs.

6. IANA Considerations

IANA does not need to take any action for this draft.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2. Informative References

[NVOPS] Narten, T. et al, "Problem Statement : Overlays for Network Virtualization", draft-narten-nvo3-overlay-problem-statement (work in progress)

[OVCPREQ] Kreeger, L. et al, "Network Virtualization Overlay Control Protocol Requirements", draft-kreeger-nvo3-overlay-cp (work in progress)

[DCVPN] Bitar, N. et al, "Cloud Networking: Framework and VPN Applicability", draft-bitar-datacenter-vpn-applicability (work in progress)

[EVPN] Raggarwa, R. et al. "BGP MPLS based Ethernet VPN", draft-ietf-l2vpn-evpn (work in progress)

[NVGRE] Sridhavan, M. et al, "NVGRE: Network Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre (work in progress)

[STT] Davie, B., "A Stateless Transport Tunneling Protocol for Network Virtualization", draft-davie-stt (work in progress)

[VXLAN] Mahalingam, M. et al, "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", draft-mahalingam-dutt-dcops-vxlan (work in progress)

[FLOYD] Sally Floyd, Allyn Romanow, "Dynamics of TCP Traffic over ATM Networks", IEEE JSAC, V. 13 N. 4, May 1995

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

[RFC1191] Mogul, J. "Path MTU Discovery", RFC1191, November 1990

[RFC1981] McCann, J. et al, "Path MTU Discovery for IPv6", RFC1981, August 1996

[RFC4821] Mathis, M. et al, "Packetization Layer Path MTU Discovery", RFC4821, March 2007

8. Acknowledgments

In addition to the authors the following people have contributed to this document:

Nabil Bitar, Verizon

Dimitrios Stiliadis, Rotem Salomonovitch, Alcatel-Lucent

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Marc Lasserre
Alcatel-Lucent
Email: marc.lasserre@alcatel-lucent.com

Florin Balus
Alcatel-Lucent
777 E. Middlefield Road
Mountain View, CA, USA 94043
Email: florin.balus@alcatel-lucent.com

Thomas Morin
France Telecom Orange
Email: thomas.morin@orange.com