

Internet Engineering Task Force (IETF)
Request for Comments: 7364
Category: Informational
ISSN: 2070-1721

T. Narten, Ed.
IBM
E. Gray, Ed.
Ericsson
D. Black
EMC
L. Fang
Microsoft
L. Kreeger
Cisco
M. Napierala
AT&T
October 2014

Problem Statement: Overlays for Network Virtualization

Abstract

This document describes issues associated with providing multi-tenancy in large data center networks and how these issues may be addressed using an overlay-based network virtualization approach. A key multi-tenancy requirement is traffic isolation so that one tenant's traffic is not visible to any other tenant. Another requirement is address space isolation so that different tenants can use the same address space within different virtual networks. Traffic and address space isolation is achieved by assigning one or more virtual networks to each tenant, where traffic within a virtual network can only cross into another virtual network in a controlled fashion (e.g., via a configured router and/or a security gateway). Additional functionality is required to provision virtual networks, associating a virtual machine's network interface(s) with the appropriate virtual network and maintaining that association as the virtual machine is activated, migrated, and/or deactivated. Use of an overlay-based approach enables scalable deployment on large network infrastructures.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see [Section 2 of RFC 5741](#).

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc7364>.

Copyright Notice

Copyright (c) 2014 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 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	4
2. Terminology	6
3. Problem Areas	6
3.1. Need for Dynamic Provisioning	6
3.2. Virtual Machine Mobility Limitations	7
3.3. Inadequate Forwarding Table Sizes	7
3.4. Need to Decouple Logical and Physical Configuration	7
3.5. Need for Address Separation between Virtual Networks	8
3.6. Need for Address Separation between Virtual Networks and ...	8
3.7. Optimal Forwarding	9
4. Using Network Overlays to Provide Virtual Networks	10
4.1. Overview of Network Overlays	10
4.2. Communication between Virtual and Non-virtualized Networks	12
4.3. Communication between Virtual Networks	12
4.4. Overlay Design Characteristics	13
4.5. Control-Plane Overlay Networking Work Areas	14
4.6. Data-Plane Work Areas	15
5. Related IETF and IEEE Work	15
5.1. BGP/MPLS IP VPNs	16
5.2. BGP/MPLS Ethernet VPNs	16
5.3. 802.1 VLANs	17
5.4. IEEE 802.1aq -- Shortest Path Bridging	17
5.5. VDP	17
5.6. ARMD	18
5.7. TRILL	18
5.8. L2VPNs	18
5.9. Proxy Mobile IP	19
5.10. LISP	19
6. Summary	19
7. Security Considerations	19
8. References	20
8.1. Normative Reference	20
8.2. Informative References	20
Acknowledgments	22
Contributors	22
Authors' Addresses	23

1. Introduction

Data centers are increasingly being consolidated and outsourced in an effort to improve the deployment time of applications and reduce operational costs. This coincides with an increasing demand for compute, storage, and network resources from applications. In order to scale compute, storage, and network resources, physical resources are being abstracted from their logical representation, in what is referred to as server, storage, and network virtualization. Virtualization can be implemented in various layers of computer systems or networks.

The demand for server virtualization is increasing in data centers. With server virtualization, each physical server supports multiple virtual machines (VMs), each running its own operating system, middleware, and applications. Virtualization is a key enabler of workload agility, i.e., allowing any server to host any application and providing the flexibility of adding, shrinking, or moving services within the physical infrastructure. Server virtualization provides numerous benefits, including higher utilization, increased security, reduced user downtime, reduced power usage, etc.

Multi-tenant data centers are taking advantage of the benefits of server virtualization to provide a new kind of hosting, a virtual hosted data center. Multi-tenant data centers are ones where individual tenants could belong to a different company (in the case of a public provider) or a different department (in the case of an internal company data center). Each tenant has the expectation of a level of security and privacy separating their resources from those of other tenants. For example, one tenant's traffic must never be exposed to another tenant, except through carefully controlled interfaces, such as a security gateway (e.g., a firewall).

To a tenant, virtual data centers are similar to their physical counterparts, consisting of end stations attached to a network, complete with services such as load balancers and firewalls. But unlike a physical data center, Tenant Systems connect to a virtual network (VN). To Tenant Systems, a virtual network looks like a normal network (e.g., providing an Ethernet or L3 service), except that the only end stations connected to the virtual network are those belonging to a tenant's specific virtual network.

A tenant is the administrative entity on whose behalf one or more specific virtual network instances and their associated services (whether virtual or physical) are managed. In a cloud environment, a tenant would correspond to the customer that is using a particular virtual network. However, a tenant may also find it useful to create multiple different virtual network instances. Hence, there is a one-

to-many mapping between tenants and virtual network instances. A single tenant may operate multiple individual virtual network instances, each associated with a different service.

How a virtual network is implemented does not generally matter to the tenant; what matters is that the service provided (Layer 2 (L2) or Layer 3 (L3)) has the right semantics, performance, etc. It could be implemented via a pure routed network, a pure bridged network, or a combination of bridged and routed networks. A key requirement is that each individual virtual network instance be isolated from other virtual network instances, with traffic crossing from one virtual network to another only when allowed by policy.

For data center virtualization, two key issues must be addressed. First, address space separation between tenants must be supported. Second, it must be possible to place (and migrate) VMs anywhere in the data center, without restricting VM addressing to match the subnet boundaries of the underlying data center network.

This document outlines problems encountered in scaling the number of isolated virtual networks in a data center. Furthermore, the document presents issues associated with managing those virtual networks in relation to operations, such as virtual network creation/deletion and end-node membership change. Finally, this document makes the case that an overlay-based approach has a number of advantages over traditional, non-overlay approaches. The purpose of this document is to identify the set of issues that any solution has to address in building multi-tenant data centers. With this approach, the goal is to allow the construction of standardized, interoperable implementations to allow the construction of multi-tenant data centers.

This document is the problem statement for the "Network Virtualization over Layer 3" (NVO3) Working Group. NVO3 is focused on the construction of overlay networks that operate over an IP (L3) underlay transport network. NVO3 expects to provide both L2 service and IP service to Tenant Systems (though perhaps as two different solutions). Some deployments require an L2 service, others an L3 service, and some may require both.

[Section 2](#) gives terminology. [Section 3](#) describes the problem space details. [Section 4](#) describes overlay networks in more detail. [Section 5](#) reviews related and further work, and [Section 6](#) closes with a summary.

2. Terminology

This document uses the same terminology as [RFC7365]. In addition, this document use the following terms.

Overlay Network: A virtual network in which the separation of tenants is hidden from the underlying physical infrastructure. That is, the underlying transport network does not need to know about tenancy separation to correctly forward traffic. IEEE 802.1 Provider Backbone Bridging (PBB) [IEEE-802.1Q] is an example of an L2 overlay network. PBB uses MAC-in-MAC encapsulation (where "MAC" refers to "Media Access Control"), and the underlying transport network forwards traffic using only the Backbone MAC (B-MAC) and Backbone VLAN Identifier (B-VID) in the outer header. The underlay transport network is unaware of the tenancy separation provided by, for example, a 24-bit Backbone Service Instance Identifier (I-SID).

C-VLAN: This document refers to Customer VLANs (C-VLANs) as implemented by many routers, i.e., an L2 virtual network identified by a Customer VLAN Identifier (C-VID). An end station (e.g., a VM) in this context that is part of an L2 virtual network will effectively belong to a C-VLAN. Within an IEEE 802.1Q-2011 network, other tags may be used as well, but such usage is generally not visible to the end station. Section 5.3 provides more details on VLANs defined by [IEEE-802.1Q].

This document uses the phrase "virtual network instance" with its ordinary meaning to represent an instance of a virtual network. Its usage may differ from the "VNI" acronym defined in the framework document [RFC7365]. The "VNI" acronym is not used in this document.

3. Problem Areas

The following subsections describe aspects of multi-tenant data center networking that pose problems for network infrastructure. Different problem aspects may arise based on the network architecture and scale.

3.1. Need for Dynamic Provisioning

Some service providers offer services to multiple customers whereby services are dynamic and the resources assigned to support them must be able to change quickly as demand changes. In current systems, it can be difficult to provision resources for individual tenants (e.g., QoS) in such a way that provisioned properties migrate automatically when services are dynamically moved around within the data center to optimize workloads.

3.2. Virtual Machine Mobility Limitations

A key benefit of server virtualization is virtual machine (VM) mobility. A VM can be migrated from one server to another live, i.e., while continuing to run and without needing to shut down and restart at the new location. A key requirement for live migration is that a VM retain critical network state at its new location, including its IP and MAC address(es). Preservation of MAC addresses may be necessary, for example, when software licenses are bound to MAC addresses. More generally, any change in the VM's MAC addresses resulting from a move would be visible to the VM and thus potentially result in unexpected disruptions. Retaining IP addresses after a move is necessary to prevent existing transport connections (e.g., TCP) from breaking and needing to be restarted.

In data center networks, servers are typically assigned IP addresses based on their physical location, for example, based on the Top-of-Rack (ToR) switch for the server rack or the C-VLAN configured to the server. Servers can only move to other locations within the same IP subnet. This constraint is not problematic for physical servers, which move infrequently, but it restricts the placement and movement of VMs within the data center. Any solution for a scalable multi-tenant data center must allow a VM to be placed (or moved) anywhere within the data center without being constrained by the subnet boundary concerns of the host servers.

3.3. Inadequate Forwarding Table Sizes

Today's virtualized environments place additional demands on the forwarding tables of forwarding nodes in the physical infrastructure. The core problem is that location independence results in specific end state information being propagated into the forwarding system (e.g., /32 host routes in IPv4 networks or MAC addresses in IEEE 802.3 Ethernet networks). In L2 networks, for instance, instead of just one address per server, the network infrastructure may have to learn addresses of the individual VMs (which could range in the hundreds per server). This increases the demand on a forwarding node's table capacity compared to non-virtualized environments.

3.4. Need to Decouple Logical and Physical Configuration

Data center operators must be able to achieve high utilization of server and network capacity. For efficient and flexible allocation, operators should be able to spread a virtual network instance across servers in any rack in the data center. It should also be possible to migrate compute workloads to any server anywhere in the network while retaining the workload's addresses.

In networks of many types (e.g., IP subnets, MPLS VPNs, VLANs, etc.), moving servers elsewhere in the network may require expanding the scope of a portion of the network (e.g., subnet, VPN, VLAN, etc.) beyond its original boundaries. While this can be done, it requires potentially complex network configuration changes and may, in some cases (e.g., a VLAN or L2VPN), conflict with the desire to bound the size of broadcast domains. In addition, when VMs migrate, the physical network (e.g., access lists) may need to be reconfigured, which can be time consuming and error prone.

An important use case is cross-pod expansion. A pod typically consists of one or more racks of servers with associated network and storage connectivity. A tenant's virtual network may start off on a pod and, due to expansion, require servers/VMs on other pods, especially the case when other pods are not fully utilizing all their resources. This use case requires that virtual networks span multiple pods in order to provide connectivity to all of the tenants' servers/VMs. Such expansion can be difficult to achieve when tenant addressing is tied to the addressing used by the underlay network or when the expansion requires that the scope of the underlying C-VLAN expand beyond its original pod boundary.

3.5. Need for Address Separation between Virtual Networks

Individual tenants need control over the addresses they use within a virtual network. But it can be problematic when different tenants want to use the same addresses or even if the same tenant wants to reuse the same addresses in different virtual networks. Consequently, virtual networks must allow tenants to use whatever addresses they want without concern for what addresses are being used by other tenants or other virtual networks.

3.6. Need for Address Separation between Virtual Networks and Infrastructure

As in the previous case, a tenant needs to be able to use whatever addresses it wants in a virtual network independent of what addresses the underlying data center network is using. Tenants (and the underlay infrastructure provider) should be able use whatever addresses make sense for them without having to worry about address collisions between addresses used by tenants and those used by the underlay data center network.

3.7. Optimal Forwarding

Another problem area relates to the optimal forwarding of traffic between peers that are not connected to the same virtual network. Such forwarding happens when a host on a virtual network communicates with a host not on any virtual network (e.g., an Internet host) as well as when a host on a virtual network communicates with a host on a different virtual network. A virtual network may have two (or more) gateways for forwarding traffic onto and off of the virtual network, and the optimal choice of which gateway to use may depend on the set of available paths between the communicating peers. The set of available gateways may not be equally "close" to a given destination. The issue appears both when a VM is initially instantiated on a virtual network or when a VM migrates or is moved to a different location. After a migration, for instance, a VM's best-choice gateway for such traffic may change, i.e., the VM may get better service by switching to the "closer" gateway, and this may improve the utilization of network resources.

IP implementations in network endpoints typically do not distinguish between multiple routers on the same subnet -- there may only be a single default gateway in use, and any use of multiple routers usually considers all of them to be one hop away. Routing protocol functionality is constrained by the requirement to cope with these endpoint limitations -- for example, the Virtual Router Redundancy Protocol (VRRP) has one router serve as the master to handle all outbound traffic. This problem can be particularly acute when the virtual network spans multiple data centers, as a VM is likely to receive significantly better service when forwarding external traffic through a local router compared to using a router at a remote data center.

The optimal forwarding problem applies to both outbound and inbound traffic. For outbound traffic, the choice of outbound router determines the path of outgoing traffic from the VM, which may be sub-optimal after a VM move. For inbound traffic, the location of the VM within the IP subnet for the VM is not visible to the routers beyond the virtual network. Thus, the routing infrastructure will have no information as to which of the two externally visible gateways leading into the virtual network would be the better choice for reaching a particular VM.

The issue is further complicated when middleboxes (e.g., load balancers, firewalls, etc.) must be traversed. Middleboxes may have session state that must be preserved for ongoing communication, and traffic must continue to flow through the middlebox, regardless of which router is "closest".

4. Using Network Overlays to Provide Virtual Networks

Virtual networks are used to isolate a tenant's traffic from that of other tenants (or even traffic within the same tenant network that requires isolation). There are two main characteristics of virtual networks:

1. Virtual networks isolate the address space used in one virtual network from the address space used by another virtual network. The same network addresses may be used in different virtual networks at the same time. In addition, the address space used by a virtual network is independent from that used by the underlying physical network.
2. Virtual networks limit the scope of packets sent on the virtual network. Packets sent by Tenant Systems attached to a virtual network are delivered as expected to other Tenant Systems on that virtual network and may exit a virtual network only through controlled exit points, such as a security gateway. Likewise, packets sourced from outside of the virtual network may enter the virtual network only through controlled entry points, such as a security gateway.

4.1. Overview of Network Overlays

To address the problems described in [Section 3](#), a network overlay approach can be used.

The idea behind an overlay is quite straightforward. Each virtual network instance is implemented as an overlay. The original packet is encapsulated by the first-hop network device, called a Network Virtualization Edge (NVE), and tunneled to a remote NVE. The encapsulation identifies the destination of the device that will perform the decapsulation (i.e., the egress NVE for the tunneled packet) before delivering the original packet to the endpoint. The rest of the network forwards the packet based on the encapsulation header and can be oblivious to the payload that is carried inside.

Overlays are based on what is commonly known as a "map-and-encap" architecture. When processing and forwarding packets, three distinct and logically separable steps take place:

1. The first-hop overlay device implements a mapping operation that determines where the encapsulated packet should be sent to reach its intended destination VM. Specifically, the mapping function maps the destination address (either L2 or L3) of a packet received from a VM into the corresponding destination address of

the egress NVE device. The destination address will be the underlay address of the NVE device doing the decapsulation and is an IP address.

2. Once the mapping has been determined, the ingress overlay NVE device encapsulates the received packet within an overlay header.
3. The final step is to actually forward the (now encapsulated) packet to its destination. The packet is forwarded by the underlay (i.e., the IP network) based entirely on its outer address. Upon receipt at the destination, the egress overlay NVE device decapsulates the original packet and delivers it to the intended recipient VM.

Each of the above steps is logically distinct, though an implementation might combine them for efficiency or other reasons. It should be noted that in L3 BGP/VPN terminology, the above steps are commonly known as "forwarding" or "virtual forwarding".

The first-hop NVE device can be a traditional switch or router or the virtual switch residing inside a hypervisor. Furthermore, the endpoint can be a VM, or it can be a physical server. Examples of architectures based on network overlays include BGP/MPLS IP VPNs [RFC4364], Transparent Interconnection of Lots of Links (TRILL) [RFC6325], the Locator/ID Separation Protocol (LISP) [RFC6830], and Shortest Path Bridging (SPB) [IEEE-802.1aq].

In the data plane, an overlay header provides a place to carry either the virtual network identifier or an identifier that is locally significant to the edge device. In both cases, the identifier in the overlay header specifies which specific virtual network the data packet belongs to. Since both routed and bridged semantics can be supported by a virtual data center, the original packet carried within the overlay header can be an Ethernet frame or just the IP packet.

A key aspect of overlays is the decoupling of the "virtual" MAC and/or IP addresses used by VMs from the physical network infrastructure and the infrastructure IP addresses used by the data center. If a VM changes location, the overlay edge devices simply update their mapping tables to reflect the new location of the VM within the data center's infrastructure space. Because an overlay network is used, a VM can now be located anywhere in the data center that the overlay reaches without regard to traditional constraints imposed by the underlay network, such as the C-VLAN scope or the IP subnet scope.

Multi-tenancy is supported by isolating the traffic of one virtual network instance from traffic of another. Traffic from one virtual network instance cannot be delivered to another instance without (conceptually) exiting the instance and entering the other instance via an entity (e.g., a gateway) that has connectivity to both virtual network instances. Without the existence of a gateway entity, tenant traffic remains isolated within each individual virtual network instance.

Overlays are designed to allow a set of VMs to be placed within a single virtual network instance, whether that virtual network provides a bridged network or a routed network.

4.2. Communication between Virtual and Non-virtualized Networks

Not all communication will be between devices connected to virtualized networks. Devices using overlays will continue to access devices and make use of services on non-virtualized networks, whether in the data center, the public Internet, or at remote/branch campuses. Any virtual network solution must be capable of interoperating with existing routers, VPN services, load balancers, intrusion-detection services, firewalls, etc., on external networks.

Communication between devices attached to a virtual network and devices connected to non-virtualized networks is handled architecturally by having specialized gateway devices that receive packets from a virtualized network, decapsulate them, process them as regular (i.e., non-virtualized) traffic, and finally forward them on to their appropriate destination (and vice versa).

A wide range of implementation approaches are possible. Overlay gateway functionality could be combined with other network functionality into a network device that implements the overlay functionality and then forwards traffic between other internal components that implement functionality such as full router service, load balancing, firewall support, VPN gateway, etc.

4.3. Communication between Virtual Networks

Communication between devices on different virtual networks is handled architecturally by adding specialized interconnect functionality among the otherwise isolated virtual networks. For a virtual network providing an L2 service, such interconnect functionality could be IP forwarding configured as part of the "default gateway" for each virtual network. For a virtual network providing L3 service, the interconnect functionality could be IP forwarding configured as part of routing between IP subnets, or it could be based on configured inter-virtual-network traffic policies.

In both cases, the implementation of the interconnect functionality could be distributed across the NVEs and could be combined with other network functionality (e.g., load balancing and firewall support) that is applied to traffic forwarded between virtual networks.

4.4. Overlay Design Characteristics

Below are some of the characteristics of environments that must be taken into account by the overlay technology.

1. Highly distributed systems: The overlay should work in an environment where there could be many thousands of access switches (e.g., residing within the hypervisors) and many more Tenant Systems (e.g., VMs) connected to them. This leads to a distributed mapping system that puts a low overhead on the overlay tunnel endpoints.
2. Many highly distributed virtual networks with sparse membership: Each virtual network could be highly dispersed inside the data center. Also, along with expectation of many virtual networks, the number of Tenant Systems connected to any one virtual network is expected to be relatively low; therefore, the percentage of NVEs participating in any given virtual network would also be expected to be low. For this reason, efficient delivery of multi-destination traffic within a virtual network instance should be taken into consideration.
3. Highly dynamic Tenant Systems: Tenant Systems connected to virtual networks can be very dynamic, both in terms of creation/deletion/power-on/power-off and in terms of mobility from one access device to another.
4. Be incrementally deployable, without necessarily requiring major upgrade of the entire network: The first-hop device (or end system) that adds and removes the overlay header may require new software and may require new hardware (e.g., for improved performance). The rest of the network should not need to change just to enable the use of overlays.
5. Work with existing data center network deployments without requiring major changes in operational or other practices: For example, some data centers have not enabled multicast beyond link-local scope. Overlays should be capable of leveraging underlay multicast support where appropriate, but not require its enablement in order to use an overlay solution.

6. Network infrastructure administered by a single administrative domain: This is consistent with operation within a data center, and not across the Internet.

4.5. Control-Plane Overlay Networking Work Areas

There are three specific and separate potential work areas in the area of control-plane protocols needed to realize an overlay solution. The areas correspond to different possible "on-the-wire" protocols, where distinct entities interact with each other.

One area of work concerns the address dissemination protocol an NVE uses to build and maintain the mapping tables it uses to deliver encapsulated packets to their proper destination. One approach is to build mapping tables entirely via learning (as is done in 802.1 networks). Another approach is to use a specialized control-plane protocol. While there are some advantages to using or leveraging an existing protocol for maintaining mapping tables, the fact that large numbers of NVEs will likely reside in hypervisors places constraints on the resources (CPU and memory) that can be dedicated to such functions.

From an architectural perspective, one can view the address-mapping dissemination problem as having two distinct and separable components. The first component consists of a back-end Network Virtualization Authority (NVA) that is responsible for distributing and maintaining the mapping information for the entire overlay system. For this document, we use the term "NVA" to refer to an entity that supplies answers, without regard to how it knows the answers it is providing. The second component consists of the on-the-wire protocols an NVE uses when interacting with the NVA.

The first two areas of work are thus: describing the NVA function and defining NVA-NVE interactions.

The back-end NVA could provide high performance, high resiliency, failover, etc., and could be implemented in significantly different ways. For example, one model uses a traditional, centralized "directory-based" database, using replicated instances for reliability and failover. A second model involves using and possibly extending an existing routing protocol (e.g., BGP, IS-IS, etc.). To support different architectural models, it is useful to have one standard protocol for the NVE-NVA interaction while allowing different protocols and architectural approaches for the NVA itself. Separating the two allows NVEs to transparently interact with different types of NVAs, i.e., either of the two architectural models described above. Having separate protocols could also allow for a

simplified NVE that only interacts with the NVA for the mapping table entries it needs and allows the NVA (and its associated protocols) to evolve independently over time with minimal impact to the NVEs.

A third work area considers the attachment and detachment of VMs (or Tenant Systems [RFC7365], more generally) from a specific virtual network instance. When a VM attaches, the NVE associates the VM with a specific overlay for the purposes of tunneling traffic sourced from or destined to the VM. When a VM disconnects, the NVE should notify the NVA that the Tenant System to NVE address mapping is no longer valid. In addition, if this VM was the last remaining member of the virtual network, then the NVE can also terminate any tunnels used to deliver tenant multi-destination packets within the VN to the NVE. In the case where an NVE and hypervisor are on separate physical devices separated by an access network, a standardized protocol may be needed.

In summary, there are three areas of potential work. The first area concerns the implementation of the NVA function itself and any protocols it needs (e.g., if implemented in a distributed fashion). A second area concerns the interaction between the NVA and NVEs. The third work area concerns protocols associated with attaching and detaching a VM from a particular virtual network instance. All three work areas are important to the development of scalable, interoperable solutions.

4.6. Data-Plane Work Areas

The data plane carries encapsulated packets for Tenant Systems. The data-plane encapsulation header carries a VN Context identifier [RFC7365] for the virtual network to which the data packet belongs. Numerous encapsulation or tunneling protocols already exist that can be leveraged. In the absence of strong and compelling justification, it would not seem necessary or helpful to develop yet another encapsulation format just for NVO3.

5. Related IETF and IEEE Work

The following subsections discuss related IETF and IEEE work. These subsections are not meant to provide complete coverage of all IETF and IEEE work related to data centers, and the descriptions should not be considered comprehensive. Each area aims to address particular limitations of today's data center networks. In all areas, scaling is a common theme as are multi-tenancy and VM mobility. Comparing and evaluating the work result and progress of each work area listed is out of the scope of this document. The

intent of this section is to provide a reference to the interested readers. Note that NVO3 is scoped to running over an IP/L3 underlay network.

5.1. BGP/MPLS IP VPNs

BGP/MPLS IP VPNs [RFC4364] support multi-tenancy, VPN traffic isolation, address overlapping, and address separation between tenants and network infrastructure. The BGP/MPLS control plane is used to distribute the VPN labels and the tenant IP addresses that identify the tenants (or to be more specific, the particular VPN/virtual network) and tenant IP addresses. Deployment of enterprise L3 VPNs has been shown to scale to thousands of VPNs and millions of VPN prefixes. BGP/MPLS IP VPNs are currently deployed in some large enterprise data centers. The potential limitation for deploying BGP/MPLS IP VPNs in data center environments is the practicality of using BGP in the data center, especially reaching into the servers or hypervisors. There may be computing workforce skill set issues, equipment support issues, and potential new scaling challenges. A combination of BGP and lighter-weight IP signaling protocols, e.g., the Extensible Messaging and Presence Protocol (XMPP), has been proposed to extend the solutions into the data center environment [END-SYSTEM] while taking advantage of built-in VPN features with its rich policy support; it is especially useful for inter-tenant connectivity.

5.2. BGP/MPLS Ethernet VPNs

Ethernet Virtual Private Networks (E-VPNs) [EVPN] provide an emulated L2 service in which each tenant has its own Ethernet network over a common IP or MPLS infrastructure. A BGP/MPLS control plane is used to distribute the tenant MAC addresses and the MPLS labels that identify the tenants and tenant MAC addresses. Within the BGP/MPLS control plane, a 32-bit Ethernet tag is used to identify the broadcast domains (VLANs) associated with a given L2 VLAN service instance, and these Ethernet tags are mapped to VLAN IDs understood by the tenant at the service edges. This means that any VLAN-based limitation on the customer site is associated with an individual tenant service edge, enabling a much higher level of scalability. Interconnection between tenants is also allowed in a controlled fashion.

VM mobility [MOBILITY] introduces the concept of a combined L2/L3 VPN service in order to support the mobility of individual virtual machines (VMs) between data centers connected over a common IP or MPLS infrastructure.

5.3. 802.1 VLANs

VLANs are a well-understood construct in the networking industry, providing an L2 service via a physical network in which tenant forwarding information is part of the physical network infrastructure. A VLAN is an L2 bridging construct that provides the semantics of virtual networks mentioned above: a MAC address can be kept unique within a VLAN, but it is not necessarily unique across VLANs. Traffic scoped within a VLAN (including broadcast and multicast traffic) can be kept within the VLAN it originates from. Traffic forwarded from one VLAN to another typically involves router (L3) processing. The forwarding table lookup operation may be keyed on {VLAN, MAC address} tuples.

VLANs are a pure L2 bridging construct, and VLAN identifiers are carried along with data frames to allow each forwarding point to know what VLAN the frame belongs to. Various types of VLANs are available today and can be used for network virtualization, even together. The C-VLAN, Service VLAN (S-VLAN), and Backbone VLAN (B-VLAN) IDs [IEEE-802.1Q] are 12 bits. The 24-bit I-SID [IEEE-802.1aq] allows the support of more than 16 million virtual networks.

5.4. IEEE 802.1aq -- Shortest Path Bridging

Shortest Path Bridging (SPB) [IEEE-802.1aq] is an overlay based on IS-IS that operates over L2 Ethernet. SPB supports multipathing and addresses a number of shortcomings in the original Ethernet Spanning Tree Protocol. Shortest Path Bridging Mac (SPBM) uses IEEE 802.1ah PBB (MAC-in-MAC) encapsulation and supports a 24-bit I-SID, which can be used to identify virtual network instances. SPBM provides multipathing and supports easy virtual network creation or update.

SPBM extends IS-IS in order to perform link-state routing among core SPBM nodes, obviating the need for bridge learning for communication among core SPBM nodes. Learning is still used to build and maintain the mapping tables of edge nodes to encapsulate Tenant System traffic for transport across the SPBM core.

SPB is compatible with all other 802.1 standards and thus allows leveraging of other features, e.g., VSI Discovery Protocol (VDP), Operations, Administration, and Maintenance (OAM), or scalability solutions.

5.5. VDP

VDP is the Virtual Station Interface (VSI) Discovery and Configuration Protocol specified by IEEE P802.1Qbg [IEEE-802.1Qbg]. VDP is a protocol that supports the association of a VSI with a port.

VDP is run between the end station (e.g., a server running a hypervisor) and its adjacent switch (i.e., the device on the edge of the network). VDP is used, for example, to communicate to the switch that a virtual machine (virtual station) is moving, i.e., designed for VM migration.

5.6. ARMD

The Address Resolution for Massive numbers of hosts in the Data center (ARMD) WG examined data center scaling issues with a focus on address resolution and developed a problem statement document [RFC6820]. While an overlay-based approach may address some of the "pain points" that were raised in ARMD (e.g., better support for multi-tenancy), analysis will be needed to understand the scaling trade-offs of an overlay-based approach compared with existing approaches. On the other hand, existing IP-based approaches such as proxy ARP may help mitigate some concerns.

5.7. TRILL

TRILL is a network protocol that provides an Ethernet L2 service to end systems and is designed to operate over any L2 link type. TRILL establishes forwarding paths using IS-IS routing and encapsulates traffic within its own TRILL header. TRILL, as originally defined, supports only the standard (and limited) 12-bit C-VID identifier. Work to extend TRILL to support more than 4094 VLANs has recently completed and is defined in [RFC7172]

5.8. L2VPNs

The IETF has specified a number of approaches for connecting L2 domains together as part of the L2VPN Working Group. That group, however, has historically been focused on provider-provisioned L2 VPNs, where the service provider participates in management and provisioning of the VPN. In addition, much of the target environment for such deployments involves carrying L2 traffic over WANs. Overlay approaches as discussed in this document are intended be used within data centers where the overlay network is managed by the data center operator rather than by an outside party. While overlays can run across the Internet as well, they will extend well into the data center itself (e.g., up to and including hypervisors) and include large numbers of machines within the data center itself.

Other L2VPN approaches, such as the Layer 2 Tunneling Protocol (L2TP) [RFC3931] require significant tunnel state at the encapsulating and decapsulating endpoints. Overlays require less tunnel state than other approaches, which is important to allow overlays to scale to hundreds of thousands of endpoints. It is assumed that smaller

switches (i.e., virtual switches in hypervisors or the adjacent devices to which VMs connect) will be part of the overlay network and be responsible for encapsulating and decapsulating packets.

5.9. Proxy Mobile IP

Proxy Mobile IP [RFC5213] [RFC5844] makes use of the Generic Routing Encapsulation (GRE) Key Field [RFC5845] [RFC6245], but not in a way that supports multi-tenancy.

5.10. LISP

LISP [RFC6830] essentially provides an IP-over-IP overlay where the internal addresses are end station identifiers and the outer IP addresses represent the location of the end station within the core IP network topology. The LISP overlay header uses a 24-bit Instance ID used to support overlapping inner IP addresses.

6. Summary

This document has argued that network virtualization using overlays addresses a number of issues being faced as data centers scale in size. In addition, careful study of current data center problems is needed for development of proper requirements and standard solutions.

This document identifies three potential control protocol work areas. The first involves a back-end NVA and how it learns and distributes the mapping information NVEs use when processing tenant traffic. A second involves the protocol an NVE would use to communicate with the back-end NVA to obtain the mapping information. The third potential work concerns the interactions that take place when a VM attaches or detaches from a specific virtual network instance.

There are a number of approaches that provide some, if not all, of the desired semantics of virtual networks. Each approach needs to be analyzed in detail to assess how well it satisfies the requirements.

7. Security Considerations

Because this document describes the problem space associated with the need for virtualization of networks in complex, large-scale, data-center networks, it does not itself introduce any security risks. However, it is clear that security concerns need to be a consideration of any solutions proposed to address this problem space.

Solutions will need to address both data-plane and control-plane security concerns.

In the data plane, isolation of virtual network traffic from other virtual networks is a primary concern -- for NVO3, this isolation may be based on VN identifiers that are not involved in underlay network packet forwarding between overlay edges (NVEs). Use of a VN identifier in the overlay reduces the underlay network's role in isolating virtual networks by comparison to approaches where VN identifiers are involved in packet forwarding (e.g., 802.1 VLANs as described in [Section 5.3](#)).

In addition to isolation, assurances against spoofing, snooping, transit modification and denial of service are examples of other important data-plane considerations. Some limited environments may even require confidentiality.

In the control plane, the primary security concern is ensuring that an unauthorized party does not compromise the control-plane protocol in ways that improperly impact the data plane. Some environments may also be concerned about confidentiality of the control plane.

More generally, denial-of-service concerns may also be a consideration. For example, a tenant on one virtual network could consume excessive network resources in a way that degrades services for other tenants on other virtual networks.

8. References

8.1. Normative Reference

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

8.2. Informative References

- [END-SYSTEM] Marques, P., Fang, L., Sheth, N., Napierala, M., and N. Bitar, "BGP-sigaled end-system IP/VPNs", Work in Progress, [draft-ietf-l3vpn-end-system-04](#), October 2014.
- [EVPN] Sajassi, A., Aggarwal, R., Bitar, N., Isaac, A., and J. Uttaro, "BGP MPLS Based Ethernet VPN", Work in Progress, [draft-ietf-l2vpn-evpn-10](#), October 2014.

- [IEEE-802.1Q]
IEEE, "IEEE Standard for Local and metropolitan area networks -- Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE 802.1Q-2011, August 2011, <<http://standards.ieee.org/getieee802/download/802.1Q-2011.pdf>>.
- [IEEE-802.1Qbg]
IEEE, "IEEE Standard for Local and metropolitan area networks -- Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks -- Amendment 21: Edge Virtual Bridging", IEEE 802.1Qbg-2012, July 2012, <<http://standards.ieee.org/getieee802/download/802.1Qbg-2012.pdf>>.
- [IEEE-802.1aq]
IEEE, "IEEE Standard for Local and metropolitan area networks -- Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks -- Amendment 20: Shortest Path Bridging", IEEE 802.1aq, June 2012, <<http://standards.ieee.org/getieee802/download/802.1aq-2012.pdf>>.
- [MOBILITY] Aggarwal, R., Rekhter, Y., Henderickx, W., Shekhar, R., Fang, L., and A. Sajassi, "Data Center Mobility based on E-VPN, BGP/MPLS IP VPN, IP Routing and NHRP", Work in Progress, [draft-raggarwa-data-center-mobility-07](#), June 2014.
- [RFC3931] Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", [RFC 3931](#), March 2005, <<http://www.rfc-editor.org/info/rfc3931>>.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", [RFC 4364](#), February 2006, <<http://www.rfc-editor.org/info/rfc4364>>.
- [RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", [RFC 5213](#), August 2008, <<http://www.rfc-editor.org/info/rfc5213>>.
- [RFC5844] Wakikawa, R. and S. Gundavelli, "IPv4 Support for Proxy Mobile IPv6", [RFC 5844](#), May 2010, <<http://www.rfc-editor.org/info/rfc5844>>.

- [RFC5845] Muhanna, A., Khalil, M., Gundavelli, S., and K. Leung, "Generic Routing Encapsulation (GRE) Key Option for Proxy Mobile IPv6", [RFC 5845](#), June 2010, <<http://www.rfc-editor.org/info/rfc5845>>.
- [RFC6245] Yegani, P., Leung, K., Lior, A., Chowdhury, K., and J. Navali, "Generic Routing Encapsulation (GRE) Key Extension for Mobile IPv4", [RFC 6245](#), May 2011, <<http://www.rfc-editor.org/info/rfc6245>>.
- [RFC6325] Perlman, R., Eastlake, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", [RFC 6325](#), July 2011, <<http://www.rfc-editor.org/info/6325>>.
- [RFC6820] Narten, T., Karir, M., and I. Foo, "Address Resolution Problems in Large Data Center Networks", [RFC 6820](#), January 2013, <<http://www.rfc-editor.org/info/rfc6820>>.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", [RFC 6830](#), January 2013, <<http://www.rfc-editor.org/info/rfc6830>>.
- [RFC7172] Eastlake, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "Transparent Interconnection of Lots of Links (TRILL): Fine-Grained Labeling", [RFC 7172](#), May 2014, <<http://www.rfc-editor.org/info/rfc7172>>.

Acknowledgments

Helpful comments and improvements to this document have come from Lou Berger, John Drake, Ilango Ganga, Ariel Hendel, Vinit Jain, Petr Lapukhov, Thomas Morin, Benson Schliesser, Qin Wu, Xiaohu Xu, Lucy Yong, and many others on the NVO3 mailing list.

Special thanks to Janos Farkas for his persistence and numerous detailed comments related to the lack of precision in the text relating to IEEE 802.1 technologies.

Contributors

Dinesh Dutt and Murari Sridharin were original co-authors of the Internet-Draft that led to the BoF that formed the NVO3 WG. That original draft eventually became the basis for this document.

Authors' Addresses

Thomas Narten (editor)
IBM
Research Triangle Park, NC
United States
EMail: narten@us.ibm.com

Eric Gray (editor)
Ericsson
EMail: eric.gray@ericsson.com

David Black
EMC Corporation
176 South Street
Hopkinton, MA 01748
United States
EMail: david.black@emc.com

Luyuan Fang
Microsoft
5600 148th Ave NE
Redmond, WA 98052
United States
EMail: lufang@microsoft.com

Lawrence Kreeger
Cisco
170 W. Tasman Avenue
San Jose, CA 95134
United States
EMail: kreeger@cisco.com

Maria Napierala
AT&T
200 S. Laurel Avenue
Middletown, NJ 07748
United States
EMail: mnapierala@att.com