Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: January 18, 2013

T. Narten, Ed.
IBM
M. Sridharan
Microsoft
D. Dutt

D. Black
EMC
L. Kreeger
Cisco

July 17, 2012

Problem Statement: Overlays for Network Virtualization
draft-narten-nvo3-overlay-problem-statement-03

Abstract

This document describes issues associated with providing multi-tenancy in large data center networks and an overlay-based network virtualization approach to addressing them. A key multi-tenancy requirement is traffic isolation, so that a tenant's traffic is not visible to any other tenant. This isolation can be achieved by assigning one or more virtual networks to each tenant such that traffic within a virtual network is isolated from traffic in other virtual networks. The primary functionality required is provisioning virtual networks, associating a virtual machine's virtual 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.

Table of Contents

1. Introduction

Server virtualization is increasingly becoming the norm 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.

Large scale 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.

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, end stations connect to a virtual network. To end stations, a virtual network looks like a normal network (e.g., providing an ethernet service), except that the only end stations connected to the virtual network are those belonging to the tenant.

A tenant is the administrative entity that is responsible for and manages a specific virtual network instance and its associated services (whether virtual or physical). In a cloud environment, a tenant would correspond to the customer that has defined and 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 matter to the tenant. It could be a pure routed network, a pure bridged network or a combination of bridged and routed networks. The key requirement is that each individual virtual network instance be isolated from other virtual network instances.

This document outlines the problems encountered in scaling the number of isolated networks in a data center, as well as the problems of managing the creation/deletion, membership and span of these networks and makes the case that an overlay based approach, where individual networks are implemented within individual virtual networks that are dynamically controlled by a standardized control plane provides a number of advantages over current approaches. The purpose of this document is to identify the set of problems 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.

Section 2 describes the problem space details. Section 3 describes network overlays in more detail and the potential work areas. Sections 4 and 5 review related and further work, while Section 6 closes with a summary.

2. Problem Details

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

2.1. Dynamic Provisioning

Cloud computing involves on-demand provisioning of resources for multi-tenant environments. A common example of cloud computing is the public cloud, where a cloud service provider offers elastic services to multiple customers over the same infrastructure. The on-demand nature of provisioning in conjunction with trusted hypervisors controlling network access by VMs can be achieved through resilient distributed network control mechanisms.

2.2. Virtual Machine Mobility Requirements

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 it down and restart it 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 licences 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 traditional data centers, servers are assigned IP addresses based on their physical location, for example based on the Top of Rack (ToR) switch for the server rack or the 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.

2.3. Span of Virtual Networks

Another use case is cross pod expansion. A pod typically consists of one or more racks of servers with its associated network and storage connectivity. Tenants may start off on a pod and, due to expansion, require servers/VMs on other pods, especially the case when tenants on the 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 tenant's servers/VMs.

2.4. Inadequate Forwarding Table Sizes in Switches

Today's virtualized environments place additional demands on the forwarding tables of switches. Instead of just one link-layer address per server, the switching infrastructure has to learn addresses of the individual VMs (which could range in the 100s per server). This is a requirement since traffic from/to the VMs to the rest of the physical network will traverse the physical network infrastructure. This places a much larger demand on the switches' forwarding table capacity compared to non-virtualized environments, causing more traffic to be flooded or dropped when the addresses in use exceeds the forwarding table capacity.

2.5. Decoupling 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. This can be achieved today by stretching VLANs (e.g., by using TRILL or SPB).

However, in order to limit the broadcast domain of each VLAN, multi-

destination frames within a VLAN should optimally flow only to those devices that have that VLAN configured. When workloads migrate, the physical network (e.g., access lists) may need to be reconfigured which is typically time consuming and error prone.

2.6. Separating Tenant Addressing from Infrastructure Addressing

It is highly desirable to be able to number the data center underlay network using whatever addresses make sense for it, without having to worry about address collisions between addresses used by the underlay and those used by tenants.

2.7. Communication Between Virtual and Traditional 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 traditional, 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). Additional identification, such as VLAN tags, could be used on the non-virtualized side of such a gateway to enable forwarding of traffic for multiple virtual networks over a common non-virtualized link.

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.

2.8. 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 Ethernet service, such interconnect functionality could be IP forwarding configured as part of the "default gateway" for each virtual network. For a virtual network

providing IP service, the interconnect functionality could be IP forwarding configured as part of the IP addressing structure of each virtual network. 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, firewall support) that is applied to traffic that is forwarded between virtual networks.

2.9. Overlay Design Characteristics

There are existing layer 2 overlay protocols in existence, but they were not necessarily designed to solve the problem in the environment of a highly virtualized data center. 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 end 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 end systems connected to any one virtual network is expected to be relatively low; Therefore, the percentage of access switches participating in any given virtual network would also be expected to be low. For this reason, efficient pruning of multi-destination traffic should be taken into consideration.

3. Highly dynamic end systems. End systems connected to virtual networks can be very dynamic, both in terms of creation/deletion/ power-on/off and in terms of mobility across the access switches.

4. Work with existing, widely deployed network Ethernet switches and IP routers without requiring wholesale replacement. The first hop switch that adds and removes the overlay header will require new equipment and/or new software.

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

3. Network Overlays

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

1. Providing network address space that is isolated from other virtual networks. The same network addresses may be used in different virtual networks on the same underlying network infrastructure.

2. Limiting the scope of frames sent on the virtual network. Frames sent by end systems attached to a virtual network are delivered as expected to other end systems on that virtual network and may exit a virtual network only through controlled exit points such as a security gateway. Likewise, frames sourced outside of the virtual network may enter the virtual network only through controlled entry points, such as a security gateway.

3.1. Limitations of Existing Virtual Network Models

Virtual networks are not new to networking. For example, VLANs are a well known construct in the networking industry. A VLAN is an L2 bridging construct that provides some of the semantics of virtual networks mentioned above: a MAC address is unique within a VLAN, but not necessarily across VLANs. Traffic sourced within a VLAN (including broadcast and multicast traffic) remains within the VLAN it originates from. Traffic forwarded from one VLAN to another typically involves router (L3) processing. The forwarding table look up operation is keyed on {VLAN, MAC address} tuples.

But there are problems and limitations with L2 VLANs. 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. A VLAN today is defined as a 12 bit number, limiting the total number of VLANs to 4096 (though typically, this number is 4094 since 0 and 4095 are reserved). Due to the large number of tenants that a cloud provider might service, the 4094 VLAN limit is often inadequate. In addition, there is often a need for multiple VLANs per tenant, which exacerbates the issue. The use of a sufficiently large VNID, present in the overlay control plane and possibly also in the dataplane would eliminate current VLAN size limitations associated with single 12-bit VLAN tags.

For IP/MPLS networks, Ethernet Virtual Private Network (E-VPN) [I-D.ietf-l2vpn-evpn] provides an emulated Ethernet service in which each tenant has its own Ethernet network over a common IP or MPLS infrastructure and 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 thirty two 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 the limit of 4096 VLANs is associated with an individual tenant service edge, enabling a much higher level of scalability. Interconnectivity between tenants is also allowed in a controlled fashion.

IP/MPLS networks also provide an IP VPN service (L3 VPN) [RFC4364] in which each tenant has its own IP network over a common IP or MPLS infrastructure and a BGP/MPLS control plane is used to distribute the tenant IP routes and the MPLS labels that identify the tenants and tenant IP routes. As with E-VPNs, interconnectivity between tenants is also allowed in a controlled fashion.

VM Mobility [I-D.raggarwa-data-center-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.

There are a number of VPN approaches that provide some if not all of the desired semantics of virtual networks. A gap analysis will be needed to assess how well existing approaches satisfy the requirements.

3.2. Benefits of Network Overlays

To address the problems described earlier, a network overlay model can be used.

The idea behind an overlay is quite straightforward. Each virtual network instance is implemented as an overlay. The original frame is encapsulated by the first hop network device. The encapsulation identifies the destination of the device that will perform the decapsulation before delivering the frame to the endpoint. The rest of the network forwards the frame based on the encapsulation header and can be oblivious to the payload that is carried inside. To avoid belaboring the point each time, the first hop network 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 VPNs [RFC4364], TRILL [RFC6325], LISP [I-D.ietf-lisp], and Shortest Path Bridging [SPB].

With the overlay, a virtual network identifier (or VNID) can be carried as part of the overlay header so that every data frame

explicitly identifies the specific virtual network the frame belongs to. Since both routed and bridged semantics can be supported by a virtual data center, the original frame carried within the overlay header can be an Ethernet frame complete with MAC addresses or just the IP packet.

The use of a sufficiently large VNID would address current VLAN limitations associated with single 12-bit VLAN tags. This VNID can be carried in the control plane. In the data plane, an overlay header provides a place to carry either the VNID, or a locally-significant identifier. In both cases, the identifier in the overlay header specifies which virtual network the data packet belongs to.

A key aspect of overlays is the decoupling of the "virtual" MAC and 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 switches at the edge of the overlay 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 regards to traditional constraints implied by L2 properties such as VLAN numbering, or the span of an L2 broadcast domain scoped to a single pod or access switch.

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 that has connectivity to both virtual network instances. Without the existence of this 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.

3.3. Overlay Networking Work Areas

There are three specific and separate potential work areas 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 frames to their proper destination. One approach is to build mapping tables entirely via learning (as is done in 802.1 networks). But to provide better scaling properties, a more

sophisticated approach is needed, i.e., the use of 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 NVE's will likely reside in hypervisors places constraints on the resources (cpu and memory) that can be dedicated to such functions. For example, routing protocols (e.g., IS-IS, BGP) may have scaling difficulties if implemented directly in all NVEs, based on both flooding and convergence time concerns. An alternative approach would be to use a standard query protocol between NVEs and the set of network nodes that maintain address mappings used across the data center for the entire overlay system.

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 "oracle" that is responsible for distributing and maintaining the mapping information for the entire overlay system. The second component consists of the on-the-wire protocols an NVE uses when interacting with the oracle.

The back-end oracle 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-oracle interaction while allowing different protocols and architectural approaches for the oracle itself. Separating the two allows NVEs to transparently interact with different types of oracles, 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 oracle for the mapping table entries it needs and allows the oracle (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 End Systems [I-D.lasserre-nvo3-framework] more generally) from a specific virtual network instance. When a VM attaches, the Network Virtualization Edge (NVE) [I-D.lasserre-nvo3-framework] associates the VM with a specific overlay for the purposes of tunneling traffic sourced from or destined to the VM. When a VM disconnects, it is removed from the overlay and the NVE effectively terminates any tunnels associated with the VM. To achieve this functionality, a standardized interaction between the NVE and hypervisor may be needed, for example in the case where the NVE resides on a separate device from the VM.

In summary, there are three areas of potential work. The first area concerns the oracle itself and any on-the-wire protocols it needs. A second area concerns the interaction between the oracle 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 a scalable, interoperable solution.

4. Related Work

4.1. IEEE 802.1aq - Shortest Path Bridging

Shortest Path Bridging (SPB) is an IS-IS based overlay for L2 Ethernets. SPB supports multi-pathing and addresses a number of shortcoming in the original Ethernet Spanning Tree Protocol. SPB-M uses IEEE 802.1ah MAC-in-MAC encapsulation and supports a 24-bit I-SID, which can be used to identify virtual network instances. SPB is entirely L2 based, extending the L2 Ethernet bridging model.

4.2. ARMD

ARMD is chartered to look at data center scaling issues with a focus on address resolution. ARMD is currently chartered to develop a problem statement and is not currently developing solutions. While an overlay-based approach may address some of the "pain points" that have been raised in ARMD (e.g., better support for multi-tenancy), an overlay approach may also push some of the L2 scaling concerns (e.g., excessive flooding) to the IP level (flooding via IP multicast). Analysis will be needed to understand the scaling tradeoffs 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.

4.3. TRILL

TRILL is an L2-based approach aimed at improving deficiencies and limitations with current Ethernet networks and STP in particular. Although it differs from Shortest Path Bridging in many architectural and implementation details, it is similar in that is provides an L2-based service to end systems. TRILL as defined today, supports only the standard (and limited) 12-bit VLAN model. Approaches to extend TRILL to support more than 4094 VLANs are currently under investigation [I-D.ietf-trill-fine-labeling]

4.4. 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 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 L2TP [RFC2661] require significant tunnel state at the encapsulating and decapsulating end points. Overlays require less tunnel state than other approaches, which is important to allow overlays to scale to hundreds of thousands of end points. It is assumed that smaller switches (i.e., virtual switches in hypervisors or the physical switches to which VMs connect) will be part of the overlay network and be responsible for encapsulating and decapsulating packets.

4.5. Proxy Mobile IP

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

4.6. LISP

LISP[I-D.ietf-lisp] 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.

4.7. Individual Submissions

Many individual submissions also look to addressing some or all of the issues addressed in this draft. Examples of such drafts are VXLAN [I-D.mahalingam-dutt-dcops-vxlan], NVGRE [I-D.sridharan-virtualization-nvgre] and Virtual Machine Mobility in L3 networks[I-D.wkumari-dcops-l3-vmmobility].

5. Further Work

It is believed that overlay-based approaches may be able to reduce the overall amount of flooding and other multicast and broadcast related traffic (e.g, ARP and ND) currently experienced within current data centers with a large flat L2 network. Further analysis is needed to characterize expected improvements.

6. Summary

This document has argued that network virtualization using L3 overlays addresses a number of issues being faced as data centers scale in size. In addition, careful consideration of a number of issues would lead to the development of interoperable implementation of virtualization overlays.

Three potential work were identified. The first involves the interaction that take place when a VM attaches or detaches from an overlay. A second involves the protocol an NVE would use to communicate with a backend "oracle" to learn and disseminate mapping information about the VMs the NVE communicates with. The third potential work area involves the backend oracle itself, i.e., how it provides failover and how it interacts with oracles in other domains.

7. Acknowledgments

Helpful comments and improvements to this document have come from Ariel Hendel, Vinit Jain, and Benson Schliesser.

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

TBD

10. Informative References

[I-D.ietf-l2vpn-evpn]
Sajassi, A., Aggarwal, R., Henderickx, W., Balus, F., Isaac, A., and J. Uttaro, "BGP MPLS Based Ethernet VPN", draft-ietf-l2vpn-evpn-01 (work in progress), July 2012.

[I-D.ietf-lisp]
Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-23 (work in progress), May 2012.

[I-D.ietf-trill-fine-labeling]
Eastlake, D., Zhang, M., Agarwal, P., Perlman, R., and D. Dutt, "TRILL: Fine-Grained Labeling", draft-ietf-trill-fine-labeling-01 (work in progress), June 2012.

[I-D.kreeger-nvo3-overlay-cp]
Black, D., Dutt, D., Kreeger, L., Sridhavan, M., and T. Narten, "Network Virtualization Overlay Control Protocol Requirements", draft-kreeger-nvo3-overlay-cp-00 (work in progress), January 2012.

[I-D.lasserre-nvo3-framework]
Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for DC Network Virtualization", draft-lasserre-nvo3-framework-03 (work in progress), July 2012.

[I-D.mahalingam-dutt-dcops-vxlan]
Sridhar, T., Bursell, M., Kreeger, L., Dutt, D., Wright, C., Mahalingam, M., Duda, K., and P. Agarwal, "VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", draft-mahalingam-dutt-dcops-vxlan-01 (work in progress), February 2012.

[I-D.raggarwa-data-center-mobility]
Aggarwal, R., Rekhter, Y., Henderickx, W., Shekhar, R., and L. Fang, "Data Center Mobility based on BGP/MPLS, IP Routing and NHRP", draft-raggarwa-data-center-mobility-03 (work in progress), June 2012.

[I-D.sridharan-virtualization-nvgre]
Sridhavan, M., Greenberg, A., Venkataramaiah, N., Wang, Y., Duda, K., Ganga, I., Lin, G., Pearson, M., Thaler, P., and C. Tumuluri, "NVGRE: Network Virtualization using Generic Routing Encapsulation", draft-sridharan-virtualization-nvgre-01 (work in progress), July 2012.

[I-D.wkumari-dcops-l3-vmmobility]
Kumari, W. and J. Halpern, "Virtual Machine mobility in L3 Networks.", draft-wkumari-dcops-l3-vmmobility-00 (work in progress), August 2011.

[RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"", RFC 2661, August 1999.

[RFC4023] Worster, T., Rekhter, Y., and E. Rosen, "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, March 2005.

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

[RFC5036] Andersson, L., Minei, I., and B. Thomas, "LDP Specification", RFC 5036, October 2007.

[RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.

[RFC5844] Wakikawa, R. and S. Gundavelli, "IPv4 Support for Proxy Mobile IPv6", RFC 5844, May 2010.

[RFC5845] Muhanna, A., Khalil, M., Gundavelli, S., and K. Leung, "Generic Routing Encapsulation (GRE) Key Option for Proxy Mobile IPv6", RFC 5845, June 2010.

[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.

[RFC6325] Perlman, R., Eastlake, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, July 2011.

[SPB] "IEEE P802.1aq/D4.5 Draft Standard for Local and Metropolitan Area Networks -- Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks, Amendment 8: Shortest Path Bridging", February 2012.

Appendix A. Change Log

A.1. Changes from -01

1. Removed Section 4.2 (Standardization Issues) and Section 5 (Control Plane) as those are more appropriately covered in and overlap with material in [I-D.lasserre-nvo3-framework] and [I-D.kreeger-nvo3-overlay-cp].

2. Expanded introduction and better explained terms such as tenant and virtual network instance. These had been covered in a section that has since been removed.

3. Added Section 3.3 "Overlay Networking Work Areas" to better articulate the three separable work components (or "on-the-wire protocols") where work is needed.

4. Added section on Shortest Path Bridging in Related Work section.

5. Revised some of the terminology to be consistent with [I-D.lasserre-nvo3-framework] and [I-D.kreeger-nvo3-overlay-cp].

A.2. Changes from -02

1. Numerous changes in response to discussions on the nvo3 mailing list, with majority of changes in Section 2 (Problem Details) and Section 3 (Network Overlays). Best to see diffs for specific text changes.

Authors' Addresses

Thomas Narten (editor)
IBM

Email: narten@us.ibm.com

Murari Sridharan
Microsoft

Email: muraris@microsoft.com

Dinesh Dutt

Email: ddutt.ietf@hobbesdutt.com

David Black
EMC

Email: david.black@emc.com

Lawrence Kreeger
Cisco

Email: kreeger@cisco.com