NETWORK OVERLAYS: AN INTRODUCTION

Network overlays dramatically increase the number of virtual subnets that can be created on a physical network, which in turn supports multitenancy and virtualization features such as VM mobility, and can speed configuration of new or existing services. We'll look at how network overlays work and examine pros and cons.

While network overlays are not a new concept, they have come back into the limelight, thanks to drivers brought on by large-scale virtualization. Several standards have been proposed to enable virtual networks to be layered over a physical network infrastructure: VXLAN, NVGRE, and SST. While each proposed standard uses different encapsulation techniques to solve current network limitations, they share some similarities. Let's look at how network overlays work in general.

Many advanced virtualization features require Layer 2 adjacency, which is the ability to exist in the same Ethernet broadcast domain. This requirement can cause broadcast domains to grow to unmanageable sizes. Prior to virtualization, network designs emphasized shrinking broadcast domains as much as possible and routing to the edge wherever possible. That's because routing is extremely scalable, and

routing to the edge can improve path utilization and alleviate dependence on Spanning Tree for loop prevention.

Now virtualization is forcing broadcast domains to grow, in part to enable features such as VM mobility. One way to do this is through the use of VLANs. The 802.1q standard defines the VLAN tag as a 12-bit space, providing for a max of 4,096 VLANs (actual implementation mileage will vary.) This is an easily reachable ceiling in multitenant environments where multiple internal or external customers will request multiple subnets.

All three proposed network overlay standards solve the scale issue by providing a much larger Virtual Network ID (VNID) space in the encapsulating packet. NVGRE and VXLAN are designed to be implemented in hardware and use a 24-bit VNID tag, which allows for 16 million virtual networks. STT uses a larger 32-bit ID. This provides for more space but would be more expensive to implement in hardware, where increased address size incurs additional cost for implementation in silicon.

Aiming for Flexibility

A need for flexibility in the data center also opens the door to network overlays. That is, the data center network needs to be flexible enough to support workloads that can move from one host to another on short notice, and for new services to be deployed rapidly.

VMs in a data center can migrate across physical servers for a variety of reasons, including a host failure or the need to distribute workloads. These moves traditionally require identical configuration of all network devices attached to clustered hosts. There is also a requirement for common configuration of upstream connecting switches in the form of VLAN trunking, and so on.

Network engineers and administrators face the same problem whether they are deploying new services or updating old ones--namely, the need to configure the network. Much of this work is manual, which limits scalability and flexibility and increases administrative overhead.

Overlay tunneling techniques alleviate this problem by providing Layer 2 connectivity independent of physical locality or underlying network design. By encapsulating traffic inside IP packets, that traffic can cross Layer 3 boundaries, removing the need for preconfigured VLANs and VLAN trunking.

These techniques provide massively scalable virtual network overlays on top of existing IP infrastructures. One of the keys to the technique is the removal of the dependence on underlying infrastructure configuration; as long as IP connectivity is available, the virtual networks operate. Additionally, all three techniques are transparent to the workload itself; the encapsulation is done behind the scenes so it is application independent.

How It Works

From a high-level perspective, all three proposed standards operate in the same way. Endpoints are assigned to a virtual network via a Virtual Network ID (VNID). These endpoints will belong to that virtual network regardless of their location on the underlying physical IP network.

In diagram 1 there are four virtual hosts connected via an IP network. Each host contains a Virtual End Point (VEP), which is a virtual switch capable of acting as the encapsulation/de-encapsulation point for the virtual networks (VNIDs.) Each host has two or more VNIDs operating on it and each workload assigned to a given VNID can communicate with other workloads in the same VNID, while maintaining separation from workloads in other VNIDs on the same or other hosts. Depending on the chosen encapsulation and configuration method, hosts that do not contain a given VNID will either never see packets destined for that VNID, or

will see them and drop them at ingress. This ensures the separation of tenant traffic.

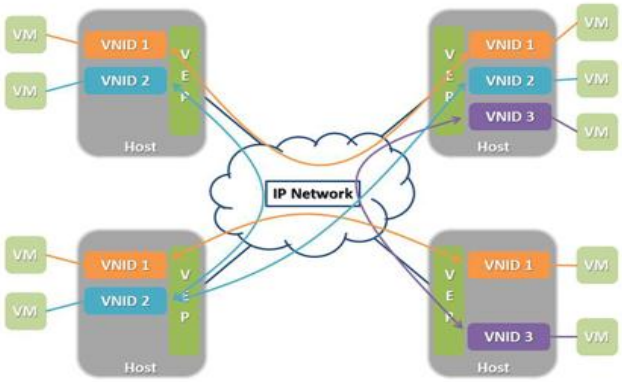
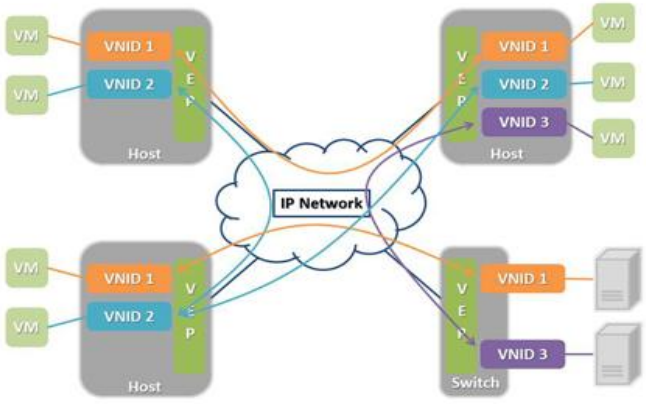


Diagram 1 focuses on virtual workloads running in VMs. The same concept would apply if using a physical switch with the VEP functionality. This would allow physical devices to be connected to the overlay network as pictured in Diagram 2 below.



With a physical switch capable of acting as the tunnel end-point, you can add both physical servers and appliances (firewalls, load balancers, and so on) to the overlay. This model is key to a cohesive deployment in mixed workload environments common in today's data centers.

Encapsulation techniques are not without drawbacks, including overhead, complications with load-balancing and interoperability issues with devices like firewalls.

The overhead with any overlay can come in two forms: encapsulation overhead of the frame size and processing overhead on the server from lack of ability to use NIC offload functionality. Both NVGRE and VXLAN suffer from the second problem due to encapsulating in IP within the soft switch. STT skirts the processing overhead problem by using a TCP hack to gain Large Segment Offload (LSO) and Large Receive Offload (LRO) capabilities from the NIC.

All three proposals will suffer from the first problem of encapsulation overhead. With any encapsulation technique you are adding additional headers to the standard frame, as shown in Diagram 3.

With modern networks the actual overhead of a few additional bytes is negligible. Where it does come into play is the size of the frame on the wire. Adding additional information will require either jumbo frame support or more fragmentation of data to meet standard frame sizes.

The three standards proposals handle this differently. VXLAN is intended to be used within a data center, where jumbo frame support is nearly ubiquitous; therefore, VXLAN assumes support and uses a larger frame size. NVGRE has provisions in the proposal for Path Maximum Transmission Unit (MTU) detection in order to use jumbo frames when possible and standard frame sizes where required. STT will be segmented by the NIC and rely on NIC settings for frame size.

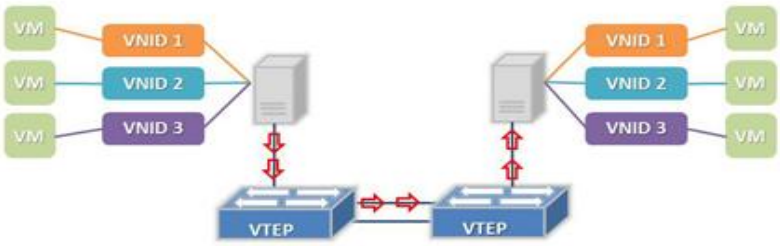
Load balancing spreads traffic across available links to maximize network throughput. It is typically done on a flow basis--that is, by device-to-device conversation. With encapsulation techniques, the inner header information becomes opaque to devices not hardware-capable of recognizing the encapsulation. This means that data normally used to provide load-balancing disappears and all communication appears as a single "flow."

VXLAN handles this issue using a hash of the of the inner payload header information as the UDP source port in the encapsulated packet. This allows for

efficient load-balancing in systems relying on 5-tuple algorithms. STT and NVGRE do not provide for as elegant of a solution, and offer up separate possibilities for providing some level of flow control.

Without a granular method of providing flow control, network traffic will bottleneck and lead to congestion that can be detrimental to the network as a whole. This will be more apparent as traffic scales up and increases the demand on network pipes.

In Diagram 4 we see all traffic from the VMs on both hosts traversing the same path, even though two are available. The same would be the case if the links were bonded such as with LACP--one physical link in the bond would always be used. This problem leaves an available link unused, and can result in performance problems if traffic overwhelms the one link being used.



The last drawback is the challenge with devices such as firewalls. These devices use header information to enforce policies and rules. Because these devices expect a specific packet format, they may be stymied by encapsulated frames. In designs

where firewalls sit in the path of encapsulated traffic, administrators will have to configure specific rules, which may be looser than traditional design.

Network overlays provide for virtualized multitenant networks on shared IP infrastructure. This provides for a more scalable design, from 4096 virtual networks to 16 million or more. In addition, a network overlay enables the flexibility and rapid provisioning required by today's business demands. Using overlays, services can be added, moved and expanded without the need for manual configuration of the underlying network infrastructure.

Source: http://www.networkcomputing.com/networking/network-overlays-an-introduction/d/d-id/1234011?