Overlay Networks in the Datacenter

Craig Johnson, Network Consulting Engineer
CCIE #6965 – Datacenter, Storage, R&S
@crajohnson
Agenda

• Why overlays?

• What problems do they solve in the datacenter

• Evolving technologies in overlay networking
Problems in Network Design

• Simplified Workload Provisioning / Automation
  • Simplified deployment
  • Fast Provisioning of Virtual Workload by consuming the L2 network from network pool
  • Without changing the physical network

• Multitenant Scale
  • Provide Layer 2 networks for tenants

• Workload anywhere (Mobility/Reachability)
  • Optimally use server resources by placing the workload anywhere
  • Yet provide Layer 2 connectivity to Workloads
Layer 2 benefits limited to a POD
Possible Solution for End-to-End L2?

Just extend STP to the whole network (!?)
Limitations of Traditional Layer 2

- Local problems have network-wide impact
- Tree topology provides limited bandwidth
  - vPC/Mlag can help a bit
- Tree topology introduces sub-optimal paths
- Flooding
- MAC address tables don’t scale
- Slow convergence
- Only 12 bit namespace for L2 domains
How network engineers want to build DC networks

Layer 3 all the way to the ToR
Advantages of a pure L3 network

• Extremely scalable
• Limited fault domain
• Localized BUM traffic
• Small MAC table sizes
• Optimized traffic flow
Cisco FabricPath Goal

FabricPath combines benefits of Layer 3 routing with simplicity of Layer 2 switching.
Limitations of Fabricpath

- Cisco Proprietary
  - TRILL went nowhere
- Improved scale over STP, but still limited
- BUM traffic still a problem
- Egress routing & ingress tromboning
Why Overlays?
Why Overlays?

Seek well integrated best in class Overlays and Underlays

Robust Underlay/Fabric
- High Capacity Resilient Fabric
- Intelligent Packet Handling
- Programmable & Manageable

Flexible Overlay Virtual Network
- Mobility – Track end-point attach at edges
- Scale – Reduce core state
  - Distribute and partition state to network edge
- Flexibility/Programmability
  - Reduced number of touch points
Types of Overlay Service

Layer 2 Overlays
- Emulate a LAN segment
- Transport Ethernet Frames (IP and non-IP)
- Single subnet mobility (L2 domain)
- Exposure to open L2 flooding
- Useful in emulating physical topologies

Layer 3 Overlays
- Abstract IP based connectivity
- Transport IP Packets
- Full mobility regardless of subnets
- Contain network related failures (floods)
- Useful in abstracting connectivity and policy

Hybrid L2/L3 Overlays offer the best of both domains
Layer 2 Overlay Considerations

• **Scale** of the edge devices
  • L2 addresses in Ethernet (MACs) use a flat space which cannot be summarized

• **L2/L3 boundary** scaling
  • Large L2 domains require a large capacity L3 gateway to handle large ARP and MAC tables at a frequent rate of refresh

• **Multi-homing** sites can induce loops in the network

• **Flooding** of L2 protocols, unknown unicasts and broadcast in general can propagate failures across the entire L2 domain
Multi-homing in L2 Overlays

Source learning assumes single attached sites
But network overlays involve edge resiliency

Enhancements are required to address:
• Loop resolution
• Multi-pathing
• Broadcast/Multicast de-duplication

Two Approaches:
• Active-Standby (Data Plane or Control Plane)
  • One active device per VLAN (single attached site)
  • VLAN based load balancing
• Active-Active (Control Plane only)
  • One active device for multi-destination traffic
  • Intra-VLAN load balancing for unicast
Flooding in L2 Overlays
Control Plane Signalling eliminates the need for floods

- Pre-set flood facility
- MAC learning based on flooding
- Flood L2 protocols and unknown unicast
  - Failure propagation
- Fail Open
- Suitable for small domains (failure scope)

- No predetermined flood tree
- MAC learning by control protocol
  - Contain Failures and L2 protocols
  - Rich information
- Fail Closed
- Better suited for broad scope
L2 Overlay Evolution

<table>
<thead>
<tr>
<th></th>
<th>VPLS</th>
<th>OTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Underlay Control</td>
<td>MPLS</td>
<td>Any IP routing</td>
</tr>
<tr>
<td>Plane</td>
<td>Flood and Learn</td>
<td>IS-IS</td>
</tr>
<tr>
<td>Encapsulation</td>
<td>MAC in MPLS</td>
<td>MAC in IP</td>
</tr>
<tr>
<td>Locator</td>
<td>MPLS PE</td>
<td>NV Edge IP</td>
</tr>
</tbody>
</table>

**Inter-DC (DCI)**

**Intra-DC (Fabric)**

**Backbone Network**

<table>
<thead>
<tr>
<th></th>
<th>Fabric Path</th>
<th>VXLAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Underlay Control</td>
<td>IS-IS</td>
<td>Any IP routing</td>
</tr>
<tr>
<td>Plane</td>
<td>Flood and Learn</td>
<td>Flood and Learn ➔ BGP</td>
</tr>
<tr>
<td>Encapsulation</td>
<td>MAC in MAC</td>
<td>MAC in IP</td>
</tr>
<tr>
<td>Locator</td>
<td>Access Switch-ID</td>
<td>Access IP</td>
</tr>
</tbody>
</table>
L2 Overlay Flood/Learn Implementations

**Fabric Path**
1. Underlay Control Plane: IS-IS calculates all possible paths between switch-IDs (Locators)
2. IS-IS calculates a multicast distribution tree for floods
3. BUM traffic flooded over multicast tree
4. Locators for each host learnt by gleaning Floods

**VXLAN**
1. Underlay Control Plane: IP calculates all possible paths between NVE-IPs (Locators)
2. IP multicast distribution tree for floods
3. BUM traffic flooded over multicast tree
4. Locators for each host learnt by gleaning Floods

**VPLS**
1. Underlay Control Plane: MPLS calculates all possible LSPs between PEs
2. Pre-determined group of pseudo-wires for flooding
3. BUM traffic flooded over multicast tree
4. PEs for each host gleaned from Floods
L2 Overlay Control Plane Implementations

1. Underlay Control Plane: IP calculates all possible paths between Edge Devices (Locators)
2. Overlay Control Plane: IS-IS adjacencies amongst Edge Devices
3. Locators for each host advertised in IS-IS
4. No Floods, integrated multi-homing

OTV

EVPN

1. Underlay Control Plane: IMPLS calculates all possible LSPs between PEs
2. Overlay Control Plane: BGP adjacencies amongst Edge Devices
3. Locators for each host advertised in BGP
4. No Floods, integrated multi-homing
Layer 3 Overlay Considerations

- **Scale** of the edge devices
  - Can be improved further by using an on-demand pull model

- **IP Mobility** for subnet disaggregation
  - Members of a subnet may be distributed across locations
  - Any host anywhere

- **Broadcast & Link-local multicast** traffic to be handled as a special case
  - Potentially without even learning MAC addresses
L3 Overlay Evolution

Edge Device Scale

**Push Protocol Model**

- IP/BGP MPLS VPNs are highly scalable today

- PE routers must:
  - Hold a large number of prefixes
  - Maintain multiple routing protocol adjacencies

- Mobility and cloud will add pressure in terms of:
  - Prefix granularity and volume
  - Increased number of PEs

**Pull Protocol (on-demand) Model**

- LISP deployments and footprint are increasing rapidly

- On-demand caching models ease the requirements on the edge devices:
  - Only prefixes being utilized are cached
  - No routing adjacencies are maintained

- A pull model is expected to provide global scalability to enable pervasive cloud models
Seminal Idea: Location and Identity Separation

Traditional Behaviour
Loc/ID “Overloaded” Semantic

10.1.0.1
Device IPv4 or IPv6 Address Represents Identity and Location

When the Device Moves, It Gets a New IPv4 or IPv6 Address for Its New Identity and Location

20.2.0.9

Overlay Behaviour
Loc/ID “Split”

10.1.0.1
Device IPv4 or IPv6 Address Represents Identity Only. Its Location Is Here!

Overlay Behaviour
Loc/ID “Split”

10.1.0.1
Device IPv4 or IPv6 Address Represents Identity Only. Its Location Is Here!

Only the Location Changes
L3 Overlay Implementations

LISP (pull)

1. Underlay Control Plane: IP calculates all possible paths between Edge Devices (Locators)
2. Overlay Control Plane: All mappings registered with Mapping System by xTRs
3. xTRs “pull” mappings on demand

MPLS VPNs (push)

1. Underlay Control Plane: IMPLS calculates all possible LSPs between PEs
2. Overlay Control Plane: BGP adjacencies amongst PEs
3. Locators for each host pushed in BGP to all PEs
Combined L2/L3 Overlays

- Route all IP traffic including Intra-subnet
- Bridge only:
  - Non-IP
  - Broadcast
  - Link-local multicast
- Assumption is that most traffic is IP
Traditional L2 - centralised L2/L3 boundary
- Always bridge, route only at an aggregation point
- Large amounts of state converge
- Scale problem for large# of L2 segments
- Traditional L2 and L2 overlays

L2/L3 fabric (or overlay)
- Always route (at the leaves), bridge when necessary
- Distribute and disaggregate necessary state
- Optimal scalability
- Enhanced forwarding and L3 overlays
IP Mobility with L3 Overlays

• Granular location information (host routes)
  • Allow subnet members to move anywhere

• Layer 2 semantics
  • ARP proxy
  • Consistent default Gateway presence

• L3 at the Access
  • Access switch replies to all ARPs with the same MAC address
  • Host routing for all traffic within the fabric
  • Summary prefix outside the fabric
L2/L3 Overlay First Hop Routing

Routing on the Leaf Nodes

• A leaf switch is assigned an IP address and a gateway MAC address for each locally defined subnet with a connected host → IP address of the SVIs

• The same anycast IP address is assigned to all leaves supporting attached hosts in the same subnet

• The same gateway MAC address can be used across all subnets supported on all the leaves
L2/L3 Overlays – ARP and Intra-subnet Forwarding

ARP Handling

1. H1 sends an ARP request for H2 – 10.1.1.20
2. The ARP request is intercepted at the leaf L1 and punted to the Sup
3. A few options:
   1. If L1 has a valid route to H2, L1 may ARP reply with its own G_MAC
   2. If L1 has a MAC-IP binding for H2, L1 may ARP-reply on behalf of H2 with H2’s MAC
   3. L1 may unicast the ARP reply to the leaf where H1 is attached
L2/L3 Overlays – ARP and Intra-subnet Forwarding
IP Forwarding within the Same Subnet

- If H1 generates a data packet destined to G_MAC, then a MAC re-write, TTL decrement and host IP forwarding takes place.
- If H1 generates a data packet destined to H2_MAC, then overlay forwarding can be done without TTL decrement based on either H2_MAC or H2_IP depending on the overlay implementation.
Combined L2/L3 Overlay Service Implementations

Cisco Nexus Unified Fabric

1. Underlay Control Plane: IS-IS calculates all possible paths between switch-IDs (Locators)
2. L2: Flood and Learn
3. L3: MP-BGP advertisement of host locations
4. Forward on L3 unless data is non-IP

Application Centric Infrastructure

1. Underlay Control Plane: IP calculates all possible paths between NVE-IPs (Locators)
2. Overlay Control Plane: COOP – Demand protocol
3. Register both IP and MACs for every host with COOP
4. Leaf nodes “pull” IP and/or MAC mappings on demand
5. Forward on L3 information unless data is non-IP
VXLAN Packet Structure

Ethernet in IP with a shim for scalable segmentation

- **Outer MAC Header**
  - Dest. Address: 48
  - Src VTEP MAC Address: 48
  - Next-Hop MAC Address: 50

- **Outer IP Header**
  - Protocol: Hex 8006
  - VLAN Type: 48
  - VLAN ID: 16
  - VLAN Tag: 16
  - Ether Type: 16

- **Outer UDP Header**
  - Source Port: 16
  - Destination Port: 16
  - Source Port: 16
  - Destination Port: 16
  - Length: 16
  - Checksum: 16

- **VXLAN Header**
  - Source Port: 32
  - Destination Port: 32
  - UDP Port: 32
  - Length: 32
  - Checksum: 32

- **Hash of the inner L2/L3/L4 headers of the original frame. Enables entropy for ECMP Load balancing in the Network.**

- **Tunnel Entropy**
  - UDP 4789
  - Allows for 16M possible segments

- **Original L2 Frame**

- **Ethernet Payload**

- **50 Bytes of overhead**

---

32
NVGRE Frame Format

NVGRE is a simple MAC-in-GRE (MAC-in-IP)

- IP header, allowing transport across any IP network
- Transport VLAN
- Allows for possible 16M segments

ProperlyEncodedFrame

Original Ethernet Frame
Overlay Taxonomy

Service = Virtual Network Instance (VNI)
Identifier = VN Identifier (VNID)
NVE = Network Virtualization Edge
VTEP = VXLAN Tunnel End-Point
VXLAN is an Overlay Encapsulation

**Data Plane Learning**
Flood and Learn over a multidestination distribution tree joined by all edge devices

**Encapsulation**

**Overlay Control Plane**

**Protocol Learning**
Advertise hosts in a protocol amongst edge devices
Data Plane Learning

Dedicated Multicast Distribution Tree per VNI

Web VM

VTEP

PIM Join for Multicast Group 239.1.1.1

DB VM

DB VM

VTEP

PIM Join for Multicast Group 239.2.2.2

Web VM

VTEP

PIM Join for Multicast Group 239.2.2.2

PIM Join for Multicast Group 239.1.1.1

Multicast-enabled Transport
Data Plane Learning

Learning on Broadcast Source - ARP Request Example

VM 1
VTEP 1
1.1.1.1

ARP Req
IP A → G

VM 2
VTEP 2
2.2.2.2

MAC
VM 1
IP Addr
VTEP 1
ARP Req

VM 3
VTEP 3
3.3.3.3

MAC
VM 1
IP Addr
VTEP 1
ARP Req

Multicast-enabled Transport
Data Plane Learning

Learning on Unicast Source - ARP Response Example

VM 1

VTEP 1
1.1.1.1

VM 2

VTEP 2
2.2.2.2

VM 3

VTEP 3
3.3.3.3

Multicast-enabled Transport
Data Plane Learning
Sharing Multicast Groups across VNIs

Blue VNI on Group G

White VNI on Group G

Purple VNI on Group G
## VXLAN L2 and L3 Gateways

Connecting VXLAN to the broader network

### L2 Gateway: VXLAN to VLAN Bridging

- Ingress VXLAN packet on Orange segment
- Egress interface chosen (bridge may .1Q tag the packet)

### L3 Gateway: VXLAN to X Routing

- VXLAN
- VLAN

- Ingress VXLAN packet on Orange segment
- Packet is routed to the new segment
- Destination is in another segment.
- Egress interface chosen (bridge may .1Q tag the packet)

### Table: VXLAN Gateway Support

<table>
<thead>
<tr>
<th></th>
<th>N1KV</th>
<th>N7K w/F3 LC</th>
<th>Nexus 3K</th>
<th>N5K/6K</th>
<th>N9K</th>
<th>CSR 1000V</th>
<th>ASR1K/ASR9K</th>
</tr>
</thead>
<tbody>
<tr>
<td>L2 Gateway</td>
<td>VXGW on 1110</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>N/A</td>
<td>Yes</td>
</tr>
<tr>
<td>L3 Gateway</td>
<td>CSR 1000V</td>
<td>Yes</td>
<td>No</td>
<td>Roadmap</td>
<td>Roadmap</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
VXLAN Evolution

Multicast Independent
- Head-end replication enables unicast-only mode
- Control Plane provides dynamic VTEP discovery

Protocol Learning prevents floods
- Workload MAC addresses learnt by VXLAN NVEs
- Advertise L2/L3 address-to-VTEP association information in a protocol

External Connectivity
- VXLAN HW Gateways to other encaps/networks
- VXLAN HW Gateway redundancy
- Enable hybrid overlays

IP Services
- VXLAN Routing
- Distributed IP Gateways
VXLAN Unicast Mode

Head-end replication

A host sends a L2 BUM* frame

1. BUM Frame

2. VTEP has static configuration of Overlay Neighbors

3. VTEP performs Head-End Replication

4. VXLAN Encap

5. Frames are unicast to the neighbors

*Broadcast, Unknown Unicast or Multicast

Overl ay Neighbors
POD3, IP C
POD2, IP B

Unicast-Only Transport
VXLAN Evolution: Using a Control Protocol

VTEP Discovery

VTEPs advertise their VNI membership in BGP

1. VTEPs advertise their VNI membership in BGP
2. BGP consolidates and propagates VTEP list for VNI
3. VTEP obtains list of VTEP neighbors for each VNI
4. VTEP can perform Head-End Replication
VXLAN Evolution: Using a Control Protocol

Protocol Learning

1. VTEPs advertise host routes (IP+MAC) to local hosts.

2. BGP propagates routes for the host to all other VTEPs.

3. VTEPs obtain host routes for remote hosts and install in RIB/FIB.

Overlay Forwarding Table:
Host1 <MAC,IP>, VTEP IP A
Host2 <MAC,IP>, VTEP IP B
Host Route Distribution decoupled from the Underlay protocol
Use MP-BGP on the leaf nodes to distribute internal host/subnet routes and external reachability information
VXLAN EVPN Control Plane

Host Advertisement

NLRI:
- Host MAC1, IP1
- NVE IP 1
- VNI 5000

Ext.Community:
- Encapsulation: VXLAN, NVGRE
- Cost/Sequence

1. Host Attaches
2. Attachment NVE advertises host’s MAC (+IP) through BGP RR
3. Choice of encapsulation is also advertised
VXLAN EVPN Control Plane

**Host Moves**

1. Host Moves to NVE3
2. NVE3 detects Host1 and advertises H1 with seq#1
3. NVE1 sees more recent route and withdraws its advertisement

**NLRI:**
- Host MAC1, IP1
- NVE IP 3
- VNI 5000

**Ext.Community:**
- Encapsulation: VXLAN, NVGRE
- Cost/Sequence 1

**MAC** | **IP** | **VNI** | **Next-Hop** | **Encap** | **Seq**
---|---|---|---|---|---
1 | 1 | 5000 | IP3 | VXLAN | 1
Overlay Control Plane IP Mobility Considerations

Host Advertisement

NLRI:
- Host MAC1, IP1
- NVE IP 1
- VNI 5000

Ext. Community:
- Encapsulation: Overlay, NVGRE
- Cost/Sequence

1. Host Attaches
2. Attachment NVE advertises host’s MAC (+IP) through BGP RR
Overlay Control Plane IP Mobility considerations

Host Moves

1. Host Moves to NVE3
2. NVE3 detects Host1 and advertises H1 with higher sequence
3. NVE1 sees more recent route and withdraws its advertisement
DC-Fabric: Normalized L2/L3 Network Overlays

- Terminate the encapsulation from the host overlay
- Translate to a normalized encapsulation in the fabric
- Seamlessly allow physical and virtual to connect to the fabric
- Fabric overlay provides L2 and L3 services with mobility and segmentation
Evolution of the VXLAN Data Plane

Beyond Ethernet in IP ➔ GPE: Generic Protocol Encapsulation

Payload:
- IP
- Ethernet
- other

50 Bytes of overhead

Source and Dst addresses of the VTEPs
Hash of the inner L2/L3/L4 headers of the original frame. Enables entropy for ECMP Load balancing in the Network.
Overlay Deployment Considerations
VTEP Redundancy in a VXLAN Fabric

vPC provides MAC state synchronization

Redundant VTEPs share anycast VTEP IP address in the underlay

vPC provides MAC state synchronization

Redundant VTEPs share anycast VTEP IP address in the underlay
Distributed IP Anycast Gateway

The same “Anycast” SVI IP/MAC is used at all VTEPs/ToRs
A host will always find its SVI anywhere it moves
Distributed IP Anycast Gateway

**Detailed View**

Consistent Anycast SVI IP / MAC address at all leaves
VLAN-IDs are locally significant
Distributed IP Anycast Gateway

Packet Flows - Pervasive VNIs/SVIs

Combined L2/L3 Overlays
• H2 ↔ H4: Bridged over VNI A
• H1 → H4: Routed via SVI B (VTEP1) → SVI A (VTEP1) → Bridged over VNI A
• H4 → H1: Routed via SVI A (VTEP2) → SVI B (VTEP2) → Bridged over VNI B
Distributed IP Anycast Gateway

Packet Flows – Scoped VNIs/SVIs

- H1 → H4: Routed via SVI B (VTEP1) → SVI A (VTEP1) → Bridged over VNI A
- H4 → H1: Routed via SVI A (VTEP2) → VNI X → SVI B (VTEP1) → VLAN B’
- Simple rule: A local route is preferred over a BGP learnt route. If there is a local SVI for the destination subnet, the SVI is the next hop
SVI/VNI/VLAN Scoping and Provisioning

Orchestration leads to scale optimization

All VNIs/SVIs everywhere
• Umbrella catch-all provisioning
• Full ARP state on all Leaf Nodes
• Can be manually provisioned up-front
• Open to L2 Flooding everywhere

VNIs/SVIs scoped as hosts attach
• Provision on host attach/policy
• ARP state only for local subnets
• Requires orchestration
• L2 Flooding is scoped
Optimizing ARP behavior

Minimizing Flooding in the Fabric with ARP suppression

• IP and MAC addresses host information distributed by control protocol
• NVEs (Leaf Switches) create an ARP cache for remote hosts
• NVEs reply locally to ARP requests for remote hosts
  • Avoid ARP request broadcast flooding

```
switch# sh ip arp suppression-cache detail

Flags: + - Adjacencies synced via CFSoE/vPC peer
  R – Remote Adjacency
  L2 – Leant over L2 interface

Total number of entries: 2
Address     Age      MAC Address      Vlan      Physical Interface  Flags
100.1.1.2   00:01:02  0026.980c.1ec2  100       Ethernet2/6          
100.1.1.3   00:01:03  0026.980c.1ec3  100       Ethernet2/6          R
```

interface nve 1
source-interface loopback 0
member vni 100 mcast-group 239.0.0.1
member vni 1000 end-host-reachability control protocol bgp
suppress-arp
ingress-replication control protocol bgp
VTEP Scaling

Selective FIB Download

- Use of /32 host routes may lead to scaling issues if all the routes are installed in the HW tables of all leaf nodes
- Host routes associated to remote endpoints can be installed into the HW FIB (from the SW RIB) upon detection of an active conversation with a local endpoint
VTEP Scaling

Pull Control Protocols

Location Data Base
Complete Host-Leaf mappings

Where is Target X?

IP Fabric

Forwarding Table
Cache only destinations for active flows

Target X is at Leaf Y

Target
Overlay Control Plane

Host and Subnet Route Distribution
Multi-Data Center Connectivity

LAN Extensions

**Domain Boundary:**
Failure and Event Containment
Clear Administrative Delineation
Multi-Data Center Connectivity

L2 Handoff

Terminate VXLAN segments at DC-edge

- Per VNI load balancing
- Optimized failure containment
VXLAN Versus LISP
Locator / ID Separation Protocol

Similarities
- Same UDP based encapsulation header
  - VXLAN does not use the control flag bits or Nonce/Map-Version field
- 24 Bit Segment ID

Differences
- LISP carries IP packets, while VXLAN carries Ethernet frames
- Forwarding Logic
  - VXLAN: Flooding/Learning
  - LISP: Uses a mapping system to register/resolve inner IP to outer IP mappings
- For LISP, IP Multicast is only required to carry host IP multicast traffic
- LISP is designed to give IP address (Identifier) mobility / multi-homing and IP core route scalability
- LISP can provide optimal traffic routing when Identifier IP addresses move to a different location
### Similarities

- Both carry Ethernet frames
- Same UDP based encapsulation header
  - VXLAN does not use the OTV Overlay ID field
- Both can use IP Multicast
  - For broadcast and multicast frames (optional for OTV)

### Differences

- **Forwarding Logic**
  - VXLAN: Flooding/Learning
  - OTV: Uses the IS-IS protocol to advertise the MAC address to IP bindings
- OTV can locally terminate ARP and doesn’t flood unknown MACs
- OTV can use an adjacency server to eliminate the need for IP multicast
- OTV is optimized for Data Center Interconnect to extend VLANs between or across data centers
- VXLAN is optimized for intra-DC and multi-tenancy
OTV & VXLAN

L2 Handoff

IP Core

Join-interface

OTV

VLAN

VXLAN

L2 internal interface

VXLAN L2 Gateway

L3 Fabric

VXLAN L2 Gateway
Multi-Data Center Connectivity

LAN Extensions and IP mobility

Ethernet extensions between independent fabrics
IP traffic is forwarded via the optimal path (no hair-pinning)
Multi-Data Center Connectivity

IP Mobility for optimized routing
End-to-end IP mobility with LISP & BGP

**L3-VXLAN & LISP IP Mobility**

**LISP Mobility:**
LISP encapsulation from client sites
No host routing in the IP core
Direct Path Forwarding

**LISP Host Mobility:**
Redistribution
BGP $\rightarrow$ LISP

**BGP Host Routes**

**Data Plane**

**Control Plane**

**MAC** | **IP** | **VNI** | **Next-Hop** | **Encap** | **Seq**
--- | --- | --- | --- | --- | ---
1 | 1 | 5000 | XTRs | VXLAN | 1

**EID** | **VNI** | **Loc** | **Seq**
--- | --- | --- | ---
1 | 5000 | DC2 | 1
Underlay Deployment Considerations
Fabric Relevance to a Hybrid Overlay

Multicast Services  
Fast Re-route

Distributed Routing Services

L2/L3 Fabric

IP Mobility Services  
L2 Services  
Overlay aware instrumentation

Overlay HW GWYs & TEPs

Site Demarcation

DCI  
WAN Integration
## Multicast Enabled Underlay

### Network Overlay

- May use PIM-ASM or PIM-BiDir (Different hardware has different capabilities)

<table>
<thead>
<tr>
<th>Mcast mode</th>
<th>N1KV</th>
<th>Nexus 7K with F3 LC</th>
<th>Nexus 3K</th>
<th>Nexus 5K/6K</th>
<th>Nexus 9K Standalone</th>
<th>CSR 1000V ASR1K</th>
<th>ASR9K</th>
</tr>
</thead>
<tbody>
<tr>
<td>IGMP v2/v3</td>
<td>PIM-ASM &amp; Bidir-PIM</td>
<td>PIM-ASM (Bidir – Future)</td>
<td>Bidir-PIM</td>
<td>PIM-ASM</td>
<td>Bidir-PIM (ASM –Future)</td>
<td>PIM-ASM &amp; Bidir-PIM</td>
<td></td>
</tr>
</tbody>
</table>

- Spine and Aggregation switches make good RP locations in clos and traditional topologies respectively
- Reserve a range of multicast channels to service the overlay and optimize for diverse VNIs
- In clos topologies with lean spine, using multiple RPs across the multiple spines and mapping different VNIs to different RPs will provide a simple load balancing measure
Multi-Pathing and Entropy

• Symmetric Underlay Network topologies facilitate ECMP routing:
  • Multi-path load balancing
  • Fast Re-convergence on link Failures

• Polarization: Encapsulated flows appear as a single flow which hashes to a single path

• Entropy in the encapsulation header to depolarize tunnels
  • Variable UDP source port in VXLAN outer header
  • Underlay must support ECMP hashing on L4 port numbers
Folded Clos Topology

Providing Topology Symmetry

- Fully Symmetric, BW rich topology, Optimized for East-West traffic
- Lean Spine does not do any VXLAN termination/gateway
- Access to other networks through border leaf block
Access/Aggregation/Core Wide BW Topology

- Fully Symmetric, BW rich topology, Optimized for North-South traffic
- VXLAN termination/gateway @ Access and Core (or Aggregation)
- Access to other networks through Core
Instrumentation and Overlay Awareness

• Infrastructure awareness of encapsulated traffic:
  • Outer/Encapsulation header
  • Overlay shim header
  • Internal/Payload header
  • Payload

• Overlay aware Switching & Routing infrastructure:
  • ACLs, QoS, Netflow

• Inspection of encapsulated traffic
Over-speed, Encapsulation & Effective Throughput

- Encapsulation adds bits to the traffic being sent
- When receiving traffic at full line rate, the encapsulated traffic will exceed the line-rate BW of the egress interface
  - Packet drops
  - Diminished effective throughput
- The uplink BW should be greater than the downlink BW to avoid congestion by encapsulation
  - This is naturally done in the network

1500 bytes/packet (10Gbps) ➞ 1542 bytes/packet (10.1 Gbps)
64 bytes/packet (10Gbps) ➞ 106 bytes/packet (10.3 Gbps)
MTU Issues: Overlay PMTUD

• Encapsulated traffic may exceed max MTU of the path

• When traffic is encapsulated with the Don’t Fragment (DF) bit set:
  • If MTU is exceeded: IGMP unreachable message (datagram-too-big) is sent back to the encapsulating NV-edge
  • Encapsulating NV-edge will lower the tunnel MTU accordingly
  • Subsequent packets from the source will trigger an ICMP unreachable message from the NV-edge back to the server (if the traffic from the source has the DF bit set)

• If the DF bit is not set, the device sensing the MTU is exceeded should attempt to fragment the traffic
Encapsulation HW Offload

Host Overlays

• Current forwarding penalty for SW encap is about 50% throughput

• STT trick leverages TCP offload engine in existing NICs
  • TCP violation, short lived workaround
  • P2P only, no routing of flows

• VXLAN/NVGRE offload on NICs
  • The way forward for host overlays
  • Disruptive, many touch points
  • Static as ASICs: headers still in flux
  • Cisco 3rd Gen VIC 2HCY14 (stateless offload)

Network Overlays

• ASIC acceleration of overlay encapsulations
  • Fast enablement of incremental functions in header reserved fields without replacing HW

• Minimal disruption at the network access
  • Manageable number of touch points

• Encapsulation Normalization

• Maximize throughput
Multi-tenancy models

Shared Mode: Segmented Overlay, Single Underlay

Parallel Mode: Segmented Overlay and Segmented Underlay
Summary recommendations & takeaways

• Optimize the location of L2 and L3 GWYs to optimize routing and minimize failure exposure
• Leverage L3 VXLAN services enabled by control protocols as the main service and L2 extensions as the exception
• Design the underlay with the VXLAN overlay in mind
• A combination of pull protocols and push protocols may render optimal scale and resiliency
• Design the network hierarchically: both the underlay as well as the overlay
• L3 Gateways are key to a sound overlay design
• Link the provisioning of the overlay and scoping of VNI to the host orchestration system for optimal scale
VXLAN EVPN Configuration
EVPN Tenant VNI Types

- **Tenant Red (VRF Red)**
  - **SVIX**
    - Layer-3 VNI X’
    - VLAN X
    - • 1 Layer-3 VNI per Tenant (VRF) for routing
    - • VNI X’ is used for routed packets
  - **SVIA**
    - Layer-2 VNI A’
    - VLAN A
    - • 1 Layer-2 VNI per Layer-2 segment
    - • Multiple Layer-2 VNIs per tenant
    - • VNI A’ and B’ are used for bridged packets
  - **SVI B**
    - Layer-2 VNI B’
    - VLAN B
Initial configuration – Per Switch

```
feature nv overlay
feature vn-segment-vlan-based
feature bgp
nv overlay evpn
```

Enable VXLAN
Enable VLAN-based VXLAN (the currently only mode)
Enable BGP
Enable EVPN control plane for VXLAN
Enable OSPF if it's chosen to be the underlay IGP routing protocol
Enable IP PIM multicast routing in the underlay network
Enable VLAN SVI interfaces if the VTEP needs to be IP gateway and route for the VXLAN VLAN IP subnet.

Other features may need to be enabled
```
feature ospf
feature pim
feature interface-vlan
```
EVPN Tenant VRF Configuration

Create VXLAN tenant VRF

```
vrf context evpn-tenant-1
  vni 39000
  rd auto
  address-family ipv4 unicast
    route-target import 39000:39000
    route-target export 39000:39000
    route-target both auto evpn
```

Create a VXLAN Tenant VRF named “evpn-tenant-1”

Layer-3 VNI for VXLAN routing within the tenant

Define VRF RD (route distinguisher)

Define VRF Route Target and import/export policies in address-family ipv4 unicast

Example to create a 2nd tenant VRF following the above steps

```
vrf context evpn-tenant-2
  vni 39010
  rd auto
  address-family ipv4 unicast
    route-target import 39010:39010
    route-target export 39010:39010
    route-target both auto evpn
```
Layer-3 VNI Per Tenant for EVPN Routing

Configure Layer-3 VNI per EVPN Tenant VRF Routing Instant

```
vlan 3900
    name 13-vni-vlan-for-tenant-1
    vn-segment 39000

interface Vlan3900
    description 13-vni-for-tenant-1-routing
    no shutdown
    vrf member evpn-tenant-1

vrf context evpn-tenant-1
    vni 39000
    rd auto
    address-family ipv4 unicast
        route-target import 39000:39000
        route-target export 39000:39000
        route-target both auto evpn
```

- Create VLAN 3900/VNI 39000 for Layer-3 VNI for tenant evpn-tenant-1 vrf routing instance
- Create the SVI interface for VLAN 3900/VNI 39000 for VXLAN routing. Put this SVI interface into the tenant VRF context
- Associate VNI 39000 with the tenant VRF evpn-tenant-1 as its routing Layer-3 VNI
EVPN Layer-3 VNI Per Tenant for Routing Instance

Configure Layer-3 VNI per EVPN Tenant VRF Routing Instant

```
vlan 3901
   name 13-vni-vlan-for-tenant-2
   vn-segment 39010

interface Vlan3901
   description 13-vni-for-tenant-2-routing
   no shutdown
   vrf member evpn-tenant-2

vrf context evpn-tenant-2
   vni 39010
   rd auto
   address-family ipv4 unicast
      route-target import 39010:39010
      route-target export 39010:39010
      route-target both auto evpn
```
EVPN Layer-2 VNI Configuration

Map VLANs to VXLAN L2 VNIs and Configure their MP-BGP EVPN Parameters

```
 vlan 200
    vn-segment 20000
vlan 210
    vn-segment 21000

evpn
    vni 20000 12
        rd auto
        route-target import auto
        route-target export auto
    vni 21000 12
        rd auto
        route-target import auto
        route-target export auto
```

Map VLAN to VXLAN VNI

Under EVPN configuration, define RD and RT import/export policies for each Layer-2 VNIs
EVPN Layer-2 Network VXLAN VLAN SVI Interface

Configure SVI interfaces for L2 VNIs, enabling anycast-gateway

interface Vlan200
no shutdown
vrf member evpn-tenant-1
ip address 20.1.1.1/8
fabric forwarding mode anycast-gateway

interface Vlan210
no shutdown
vrf member evpn-tenant-2
ip address 21.1.1.1/8
fabric forwarding mode anycast-gateway
**EVPN Layer-2 Network VXLAN VLAN SVI Interface**

Configure SVI interfaces for L2 VNIs, enabling anycast-gateway

```plaintext
interface Vlan200
  no shutdown
  vrf member evpn-tenant-1
  ip address 20.1.1.1/8

interface Vlan210
  no shutdown
  vrf member evpn-tenant-2
  ip address 21.1.1.1/8
```

Create SVI interface vlan 200
Associate it with tenant VRF evpn-tenant-1

Create SVI interface vlan 210
Associate it with tenant VRF evpn-tenant-2
EVPN Distributed Gateway

fabric forwarding anycast-gateway-mac 0002.0002.0002

interface Vlan200
  no shutdown
  vrf member evpn-tenant-1
  ip address 20.1.1.1/8
  fabric forwarding mode anycast-gateway

interface Vlan210
  no shutdown
  vrf member evpn-tenant-2
  ip address 21.1.1.1/8
  fabric forwarding mode anycast-gateway

Configure distributed gateway virtual MAC address
One virtual MAC per VTEP.
All VTEPs should have the same virtual MAC address

Configure anycast-gateway virtual IP address for VLAN 200.
All VTEPs for VLAN 200 should have the same virtual IP address

Enable anycast-gateway fabric forwarding for VLAN 200

Configure anycast-gateway virtual IP address for VLAN 210.
All VTEPs for VLAN 210 should have the same virtual IP address

Enable anycast-gateway fabric forwarding for VLAN 210
VXLAN Tunnel Interface Configuration

Configure VXLAN tunnel interface nve1

```
interface nve1
  no shutdown
  source-interface loopback0
  host-reachability protocol bgp
  member vni 20000
    suppress-arp
    mcast-group 239.1.1.1
  member vni 21000
    suppress-arp
    mcast-group 239.1.1.2
  member vni 39000 associate-vrf
  member vni 39010 associate-vrf
```

Specify loopback0 as the source interface
Define BGP as the mechanism for host reachability advertisement
Associate tenant Layer-2 VNIs to the tunnel interface nve1
Define the mcast group on a per-VNI basis
Enable arp suppression on a per-VNI basis
Add Layer-3 VNIs, one per tenant VRF

```
interface loopback0
  ip address 10.1.1.11/32
  ip ospf network point-to-point
  ip router ospf 1 area 0.0.0.0
  ip pim sparse-mode
```

Loopback0 interface configuration
The /32 IP address is used as the VTEP address
**MP-BGP Configuration on VTEP**

```
router bgp 100
  router-id 10.1.1.11
  log-neighbor-changes
  address-family ipv4 unicast
  address-family l2vpn evpn
  neighbor 10.1.1.1 remote-as 100
    update-source loopback0
    address-family ipv4 unicast
    address-family l2vpn evpn
    send-community extended
  neighbor 10.1.1.2 remote-as 100
    update-source loopback0
    address-family ipv4 unicast
    address-family l2vpn evpn
    send-community extended

vrf evpn-tenant-1
  address-family ipv4 unicast
  advertise l2vpn evpn

vrf evpn-tenant-2
  address-family ipv4 unicast
  advertise l2vpn evpn
```

- **Address-family ipv4 unicast for prefix-based routing**
- **Address-family ipv4 evpn for vxlan host-based routing**
- **Define MP-BGP neighbors. Under each neighbor define address-family ipv4 unicast and l2vpn evpn parameters**
- **Send extended community for both MAC and Host IP reachability information**
- **Under address-family ipv4 unicast of each tenant VRF instance, enable advertising EVPN routes**
MP-BGP Configuration on iBGP Route Reflector

```
router bgp 100
  router-id 10.1.1.1
  log-neighbor-changes
  address-family ipv4 unicast
  address-family l2vpn evpn
    retain route-target all
  template peer vtep-peer
    remote-as 100
    update-source loopback0
    address-family ipv4 unicast
      send-community both
      route-reflector-client
    address-family l2vpn evpn
      send-community both
      route-reflector-client
  neighbor 10.1.1.11
    inherit peer vtep-peer
  neighbor 10.1.1.12
    inherit peer vtep-peer
  neighbor 10.1.1.13
    inherit peer vtep-peer
  neighbor 10.1.1.14
    inherit peer vtep-peer
```

- Address-family ipv4 unicast for prefix-based routing
- Address-family ipv4 evpn for vxlan host-based routing
- iBGP RR client peer template
- Ssnd both standard and extended community in address-family ipv4 unicast
- Ssnd both standard and extended community in address-family l2vpn evpn
ip pim ssm range 232.0.0.0/8
ip pim bsr listen forward

interface loopback0
   ip address 10.1.1.11/32
   ip ospf network point-to-point
   ip router ospf 1 area 0.0.0.0
   ip pim sparse-mode

interface Ethernet2/1
   description TME-1-9508-1 Eth1/1
   no switchport
   ip address 192.167.11.2/30
   ip ospf network point-to-point
   ip router ospf 1 area 0.0.0.0
   ip ospf bfd
   ip pim sparse-mode
   no shutdown

interface Ethernet2/2
   description TME-1-9508-2 Eth1/1
   no switchport
   ip address 192.168.11.2/30
   ip ospf network point-to-point
   ip router ospf 1 area 0.0.0.0
   ip ospf bfd
   ip pim sparse-mode
   no shutdown

This example uses BSR for RP