Welcome!

2025 Arista EVPN Workshop

Workshop Agenda Overview Day 1

JP

10:15am - 12:15am - Network Architectures

- DC/Campus Architectures
  - Fundamental network operation review - Pat
  - MLAG (LACP) and vARP (First hop redundancy) - Nate
  - L2LS vs L3LS vs L3LS-V - Nate
  - Underlay and Overlay review -Steve

12:15 pm - 1:00 pm - Lunch

- FOOD!

1:00pm - 4:00pm - More fun

- VXLAN Fundamentals and Operations - Steve
- Configuration Overview - vxlan1 interface - Steve
- ARP Packet Walk - Steve
  - L2LS vs. VXLAN
- Troubleshooting Basic VXLAN - Jason Hardy

Workshop Agenda Overview Day 2

JP

10:15am - 12:15am - Recap on Day 1 - Why EVPN (again?) - Use cases

- EVPN VXLAN
  - MBGP
    - What is an address family?
    - Common EVPN Route types - 2,3,5
  - Underlay/Overlay (again) - BGP for both?
- L2EVPN & L3EVPN

12:15 pm - 1:00 pm - Lunch

- FOOD!

1:00pm - 4:00pm - Even more fun!

- Routing Models
  - External Device (Firewall, Router)
  - IRB options
    - Centralized IRB
    - Symmetric IRB
- Troubleshooting EVPN
- Labs

Advanced EVPN Workshop (in the works…)

JP

- DCI
- Multicast
- Dual Homing (Arista Active/Active vs MLAG)

- WAN/Cloud
- VLAN translation
- Firewall Topologies

- RFC 5549 (EVPN Fabric Autoconfig via IPv4 over IPv6)

Arista Rockies Workshop Series

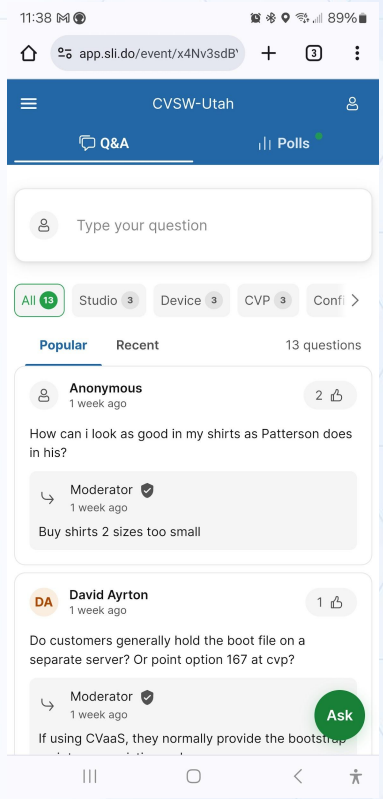
2024-2025

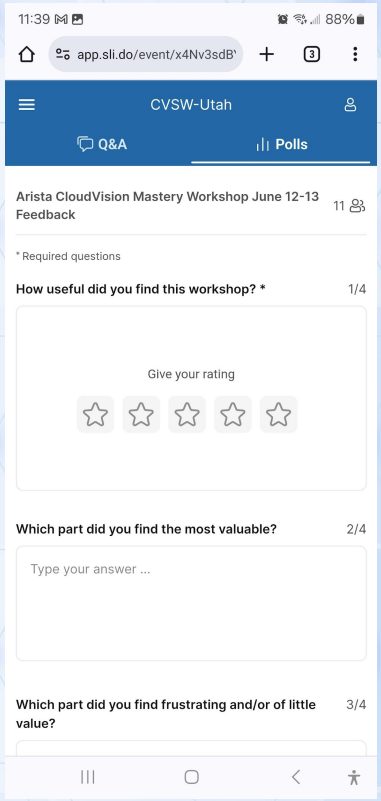
- Automation
- Cloudvision Mastery
- Campus (Wired/WiFi/AAA Policy)
- Observability (Tapp Aggregation)
- EVPN Deepdive

2026 ???

- Advanced EVPN Deepdive
- Cloudvision Mastery 2026 update?
- SD-WAN?
- ZTN?
- Other?

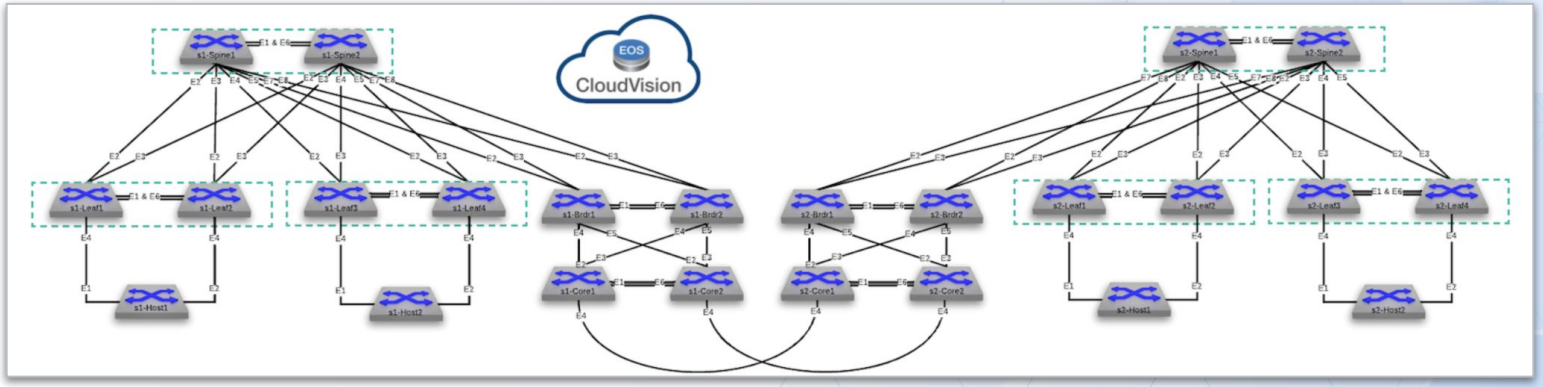
Slido: Q/A and Survey





Lab details:

- You will each receive:
  - A unique Arista Test Drives (ATD) link to your personal lab
  - Included in that link will be:
    - A virtual environment consisting of the following network topology:



- Other included items with your ATD:
  - Lab Credentials for each device in your lab, including CloudVision
  - Console and SSH access to each router
  - Lab Guide

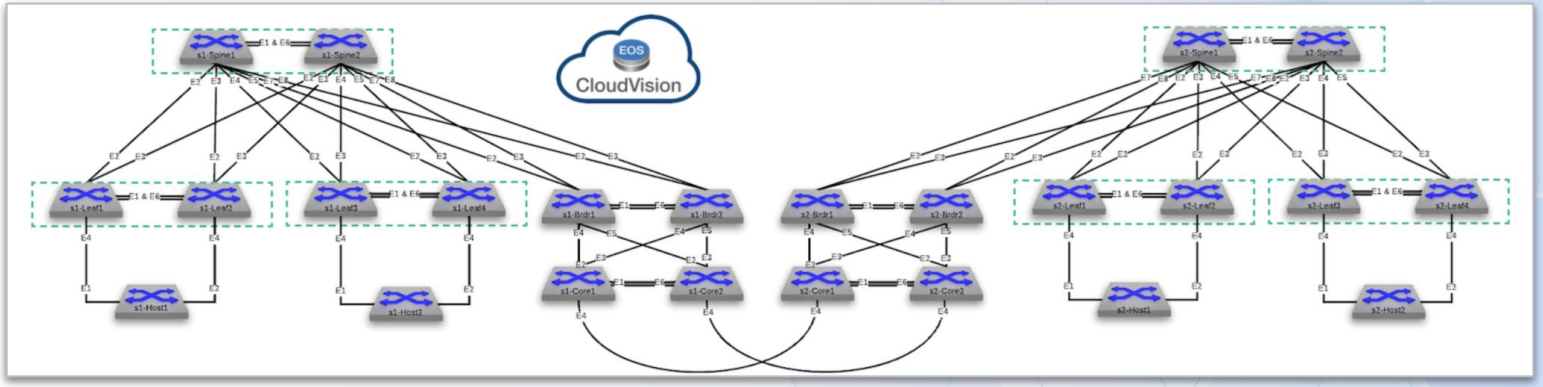
Lab Assignment

https://testdrive.arista.com/labs?lab=surely-major-crab

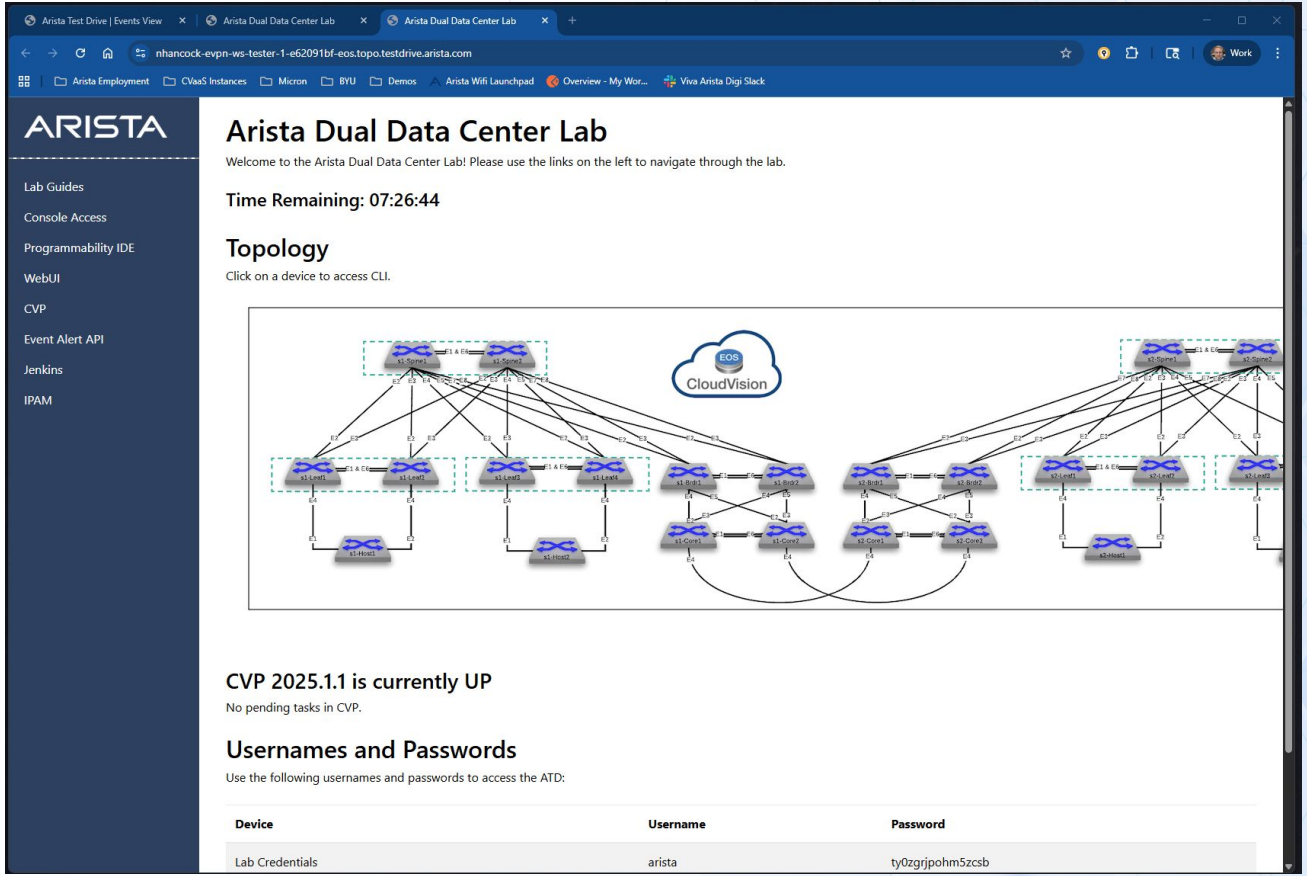
https://testdrive.arista.com/labs?lab=gently-prompt-gannet

Lab details:

- Jump into your lab!
  - Access to the CLI
  - CloudVision
  - Set Lab state



Setup lab for L2LS



ARISTA

Let’s talk Ethernet - Building a common base

PF

wikipedia

wikipedia

ARISTA

Switch and Router Basics

Have I got a CAM for you!

- What is CAM?
- What's contained in CAM?

MAC Address Table

- Fields (DA, SA, Port, Timer)
- **Operations****

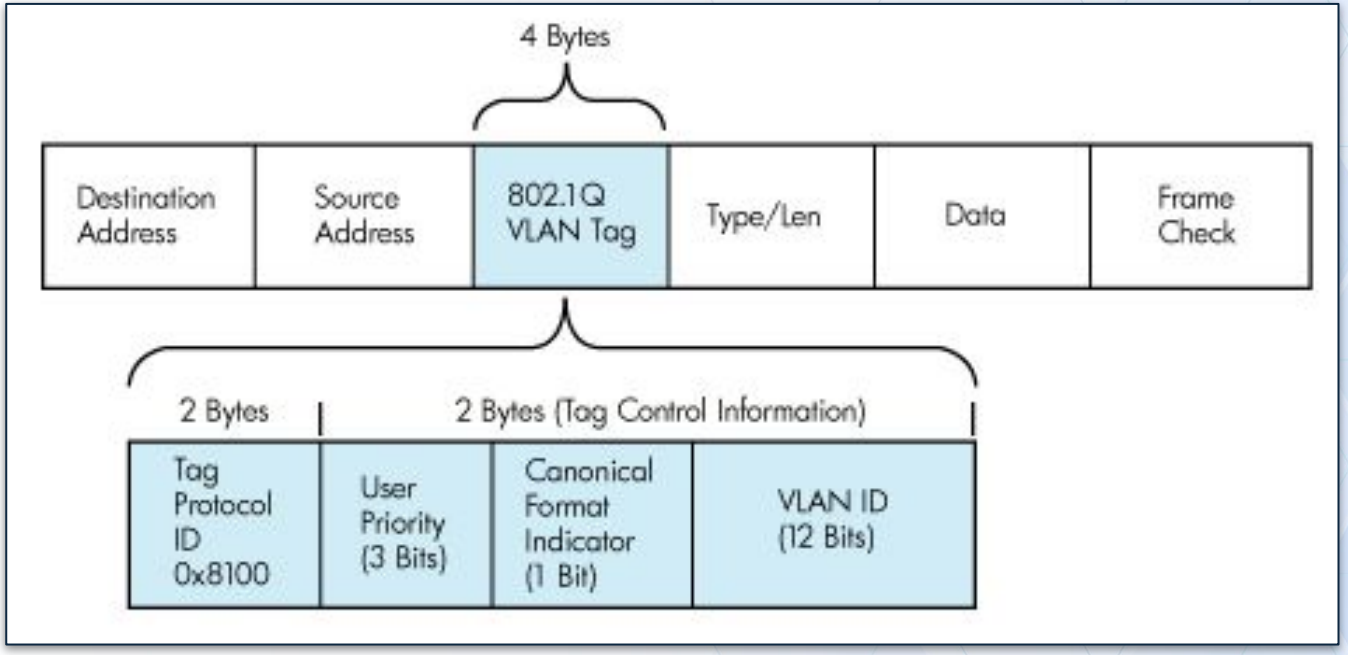
ARP Table

- Fields (IP, MAC, Port, Timer)
- **Operations****

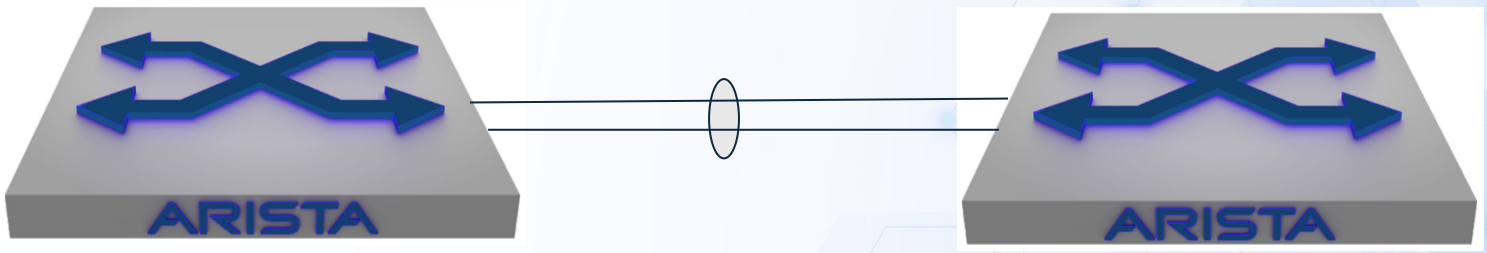
ebay

Fundamental Review: Frame diagrams - a quick look

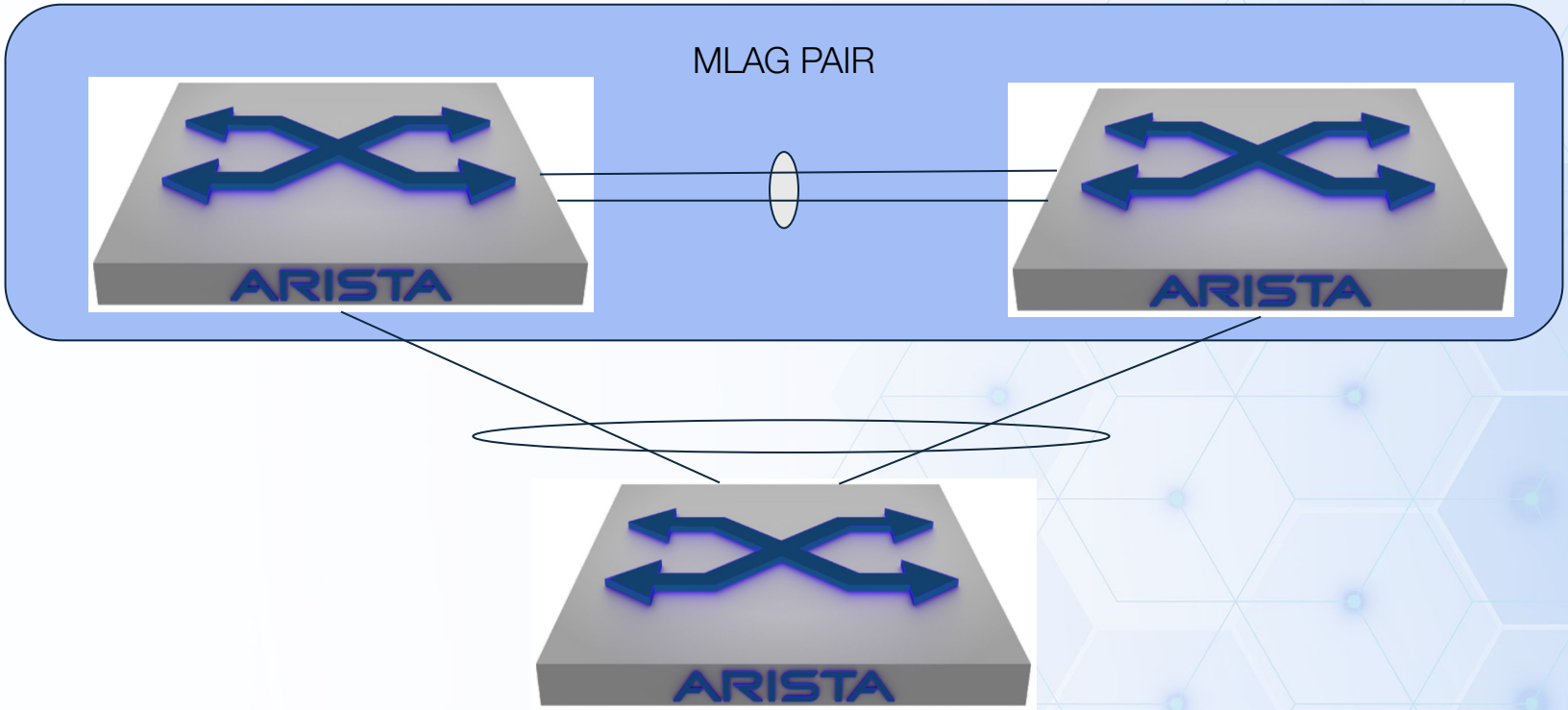
PF



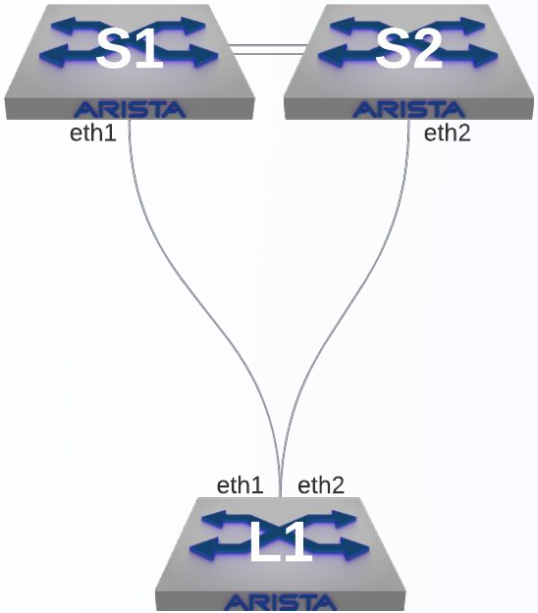
Fundamental Review: MLAG



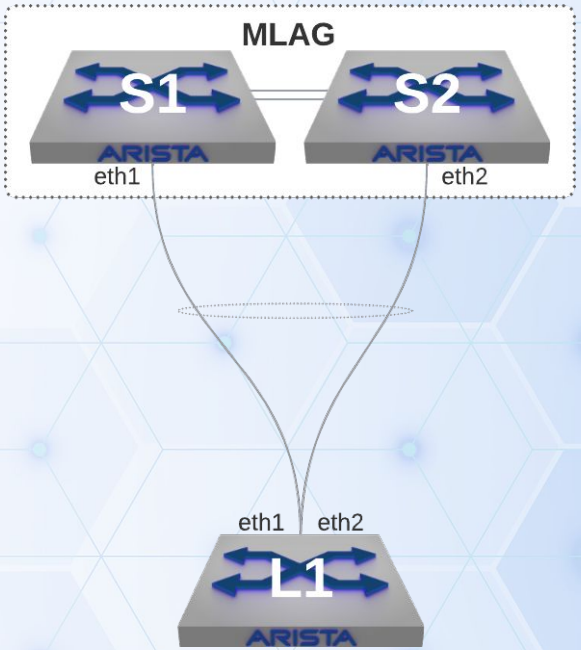
Fundamental Review: MLAG



Fundamental Review:
What does MLAG stand for and what does it do?



What are differences between this?



And this?

Pause and Test: MLAG

Try in your Lab!

Show mlag status and config

- s1-spine1#show int status
- s1-spine1#show mlag
- s1-spine1#show mlag interface (port channels that have mlag turn on)
- s1-spine1#show mlag config-sanity
- s1-spine1#show port-channel dense
- s1-spine1#show interface po12

Pause and Test: MLAG Config

NH

```
spanning-tree mode mstp
no spanning-tree vlan-id 4094

...
vlan 4094
   name MLAG-Peer
   trunk group MLAG
...

!
interface Port-Channel1
   description Peer-Link
   switchport mode trunk
   switchport trunk group MLAG
!

interface Vlan4094
   description MLAG Peering
   ip address 10.255.255.1/30
!
```

```
mlag configuration
   domain-id MLAG
   local-interface Vlan4094
   peer-address 10.255.255.2
   peer-link Port-Channel1
!
!
!

int ethernet 1
  description lacp-member-link-to-server
  channel-group 1 mode active

int ethernet 2
  description lacp-member-link-to-server
  channel-group 1 mode active

interface port-channel 1
  description LACP bond to server
  switchport mode trunk
  switchport trunk allowed vlan 10,20,30
  mlag 1
```


Pause and Test: vARP within MLAG

Try in your Lab!

- Show varp config

  s1-spine1#show run int vlan112

  s1-spine1#show run | inc mac-address

- Live pcap showing GARP on spine1

  s1-spine1# bash

  [arista@s1-spine1 ~]$ tcpdump -i vlan112 arp

- ARP behaviour from spine1 and spine2 when host1 pings gateway
  - Open 1st session to s1-host1
    - s1-host1#bash
    - [arista@s1-host1 ~]$ tcpdump -i po1 not stp
  - Open 2nd session to s1-host1
    - s1-host1#clear arp 10.111.112.1
    - s1-host1#ping 10.111.112.1

Pause and Test: vARP within MLAG


```
SPINE1#config

ip virtual-router mac-address 00:1c:73:00:00:12

interface Vlan112
   description staff-wired
   ip address 10.111.112.2/24
   ip virtual-router address 10.111.112.1
```

```
SPINE2#config

ip virtual-router mac-address 00:1c:73:00:00:12

interface Vlan112
   description staff-wired
   ip address 10.111.112.3/24
   ip virtual-router address 10.111.112.1
```


Switch from L2LS to HER Jumphost option 5

DC/Campus Architectures:

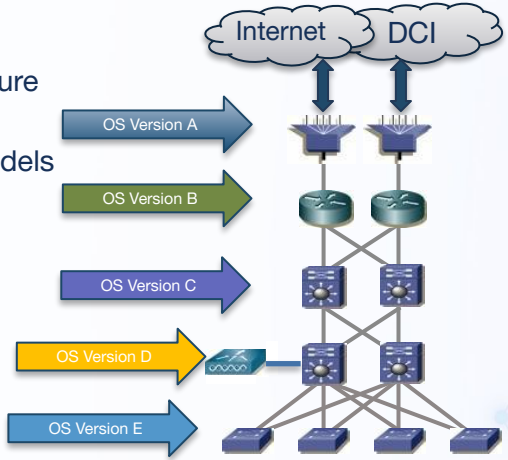
NH

Different architectural approaches

Legacy - Complex

Fragmented:

- Myriad OSes/Features
- “Good Enough” Architecture
- Incongruent Platforms
- Disparate Operational Models
- Proven Fragility

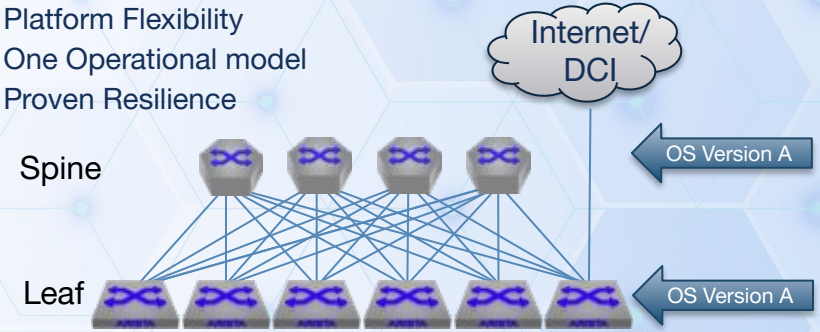


2000’s Era: Fragmented

Modern - Simplified

Consistency:

- Single OS/Features
- Deterministic Architecture
- Platform Flexibility
- One Operational model
- Proven Resilience



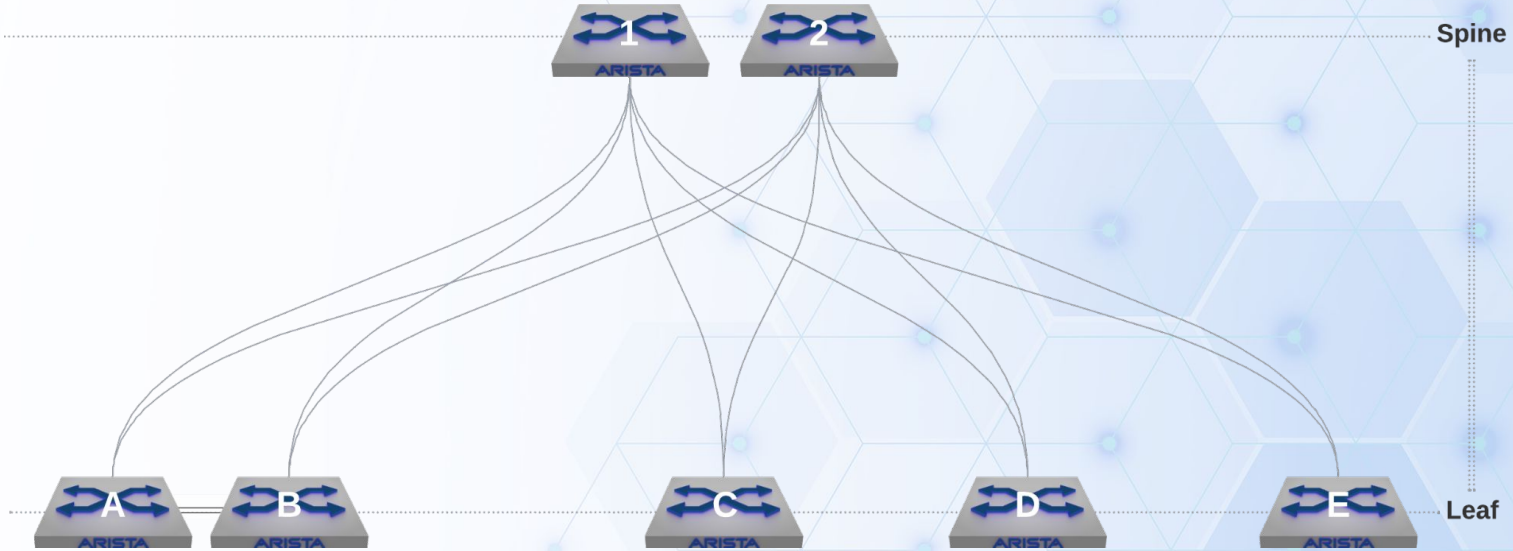
Cloud Principles

ARISTA

Leaf/Spine Architecture:

Attributes

- Structurally consistent
- Shortened East-West paths
- Reduced latency
- Scales very well
- Repeatable design

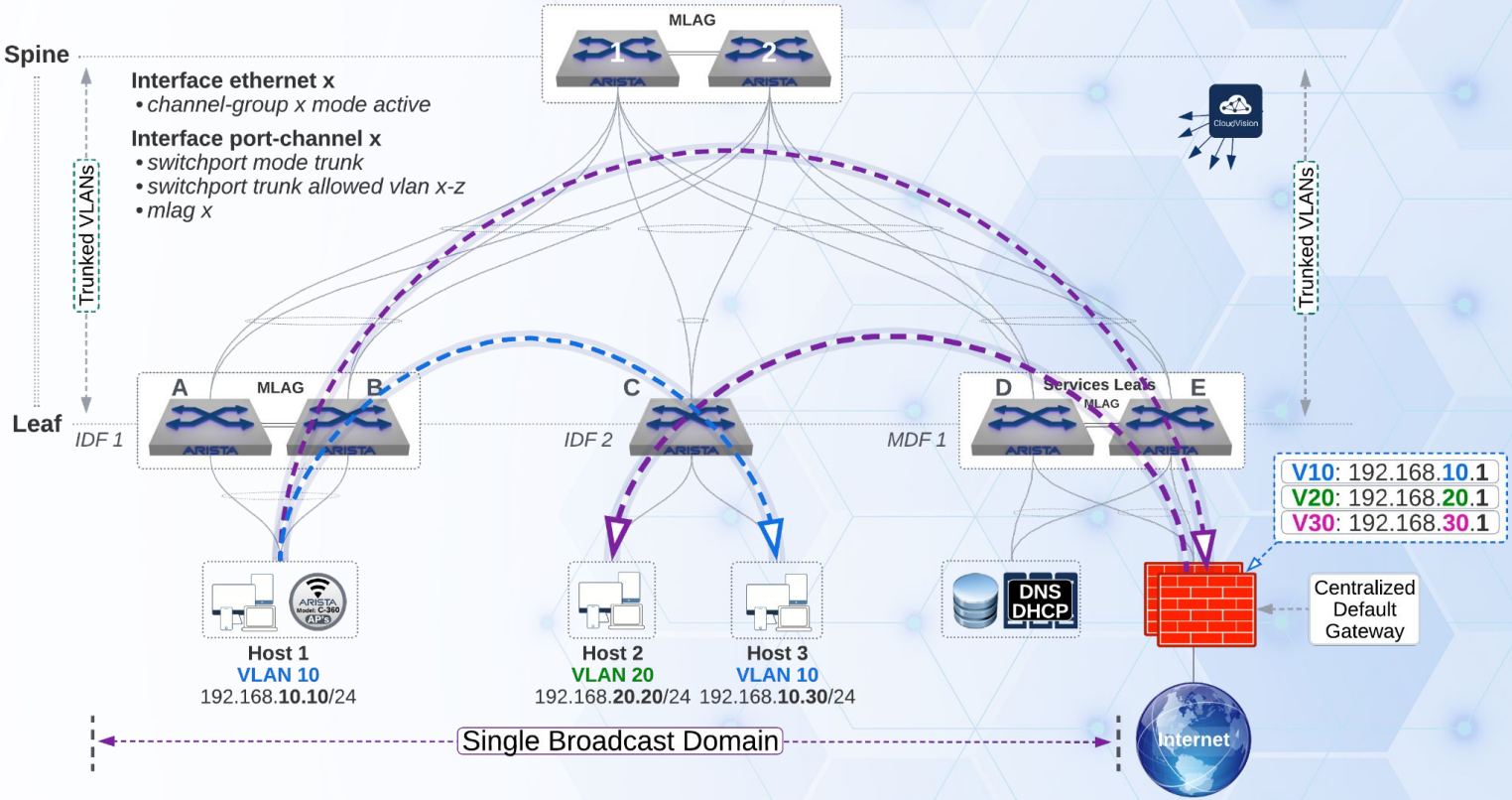


Layer 2, Leaf/Spine (L2LS):

NH

Attributes

- Tried and true
- Familiar technologies
- Links between Spine and Leaf are L2 switch ports
- Switch redundancy is provided by MLAG
- Link redundancy is provided by LACP
- Routing is usually either provided by the Spines or Firewalls or both

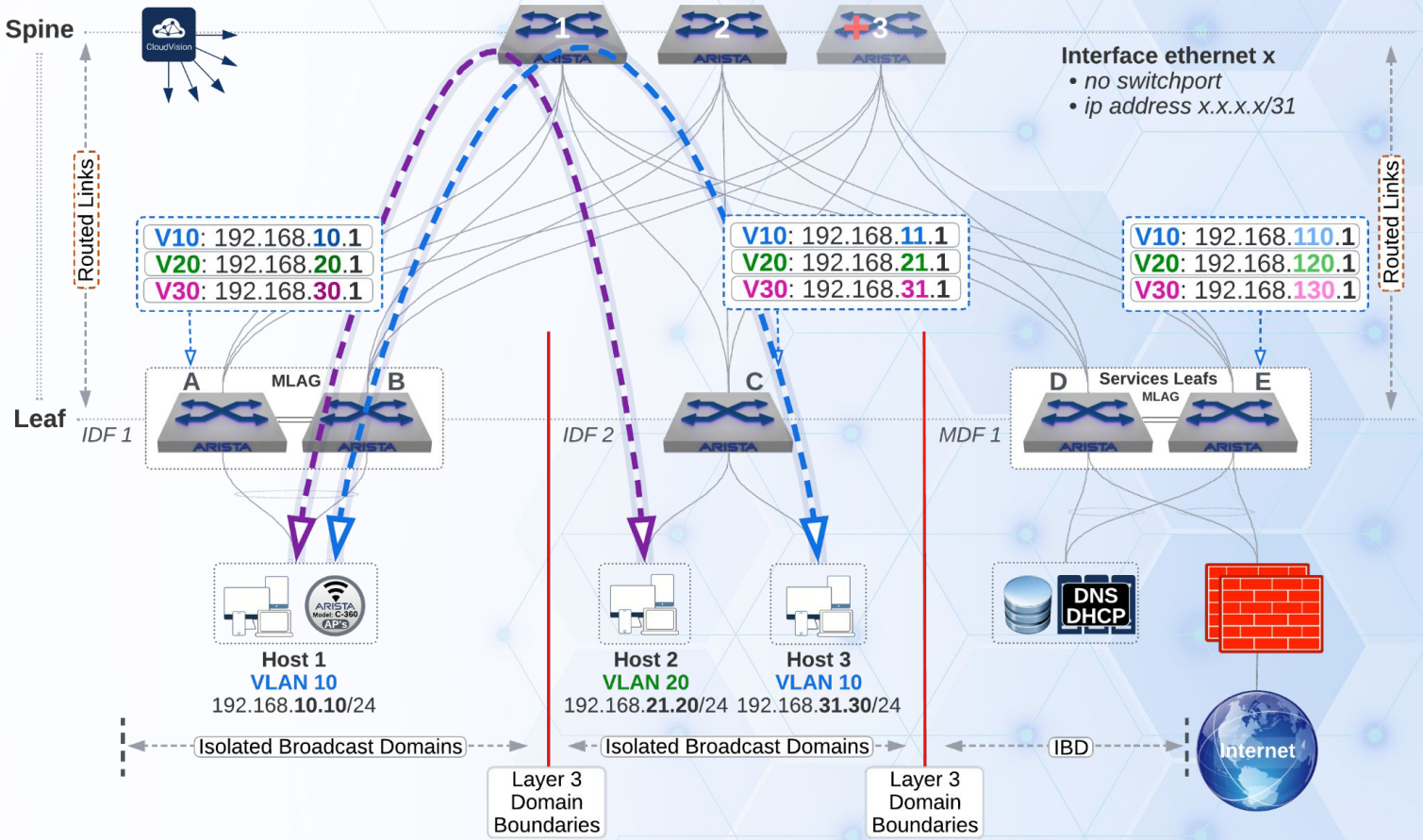


Layer 3, Leaf/Spine (L3LS):

NH

Attributes

- Links between Spine and Leaf are L3 routed with /31 subnets
- No Link Aggregation technologies are involved with the infrastructure links
- Multi-pathing provided by ECMP
- Traffic routes from Leafs to Spines
- Leafs are configured with gateways for vlans
- VRF-Lite

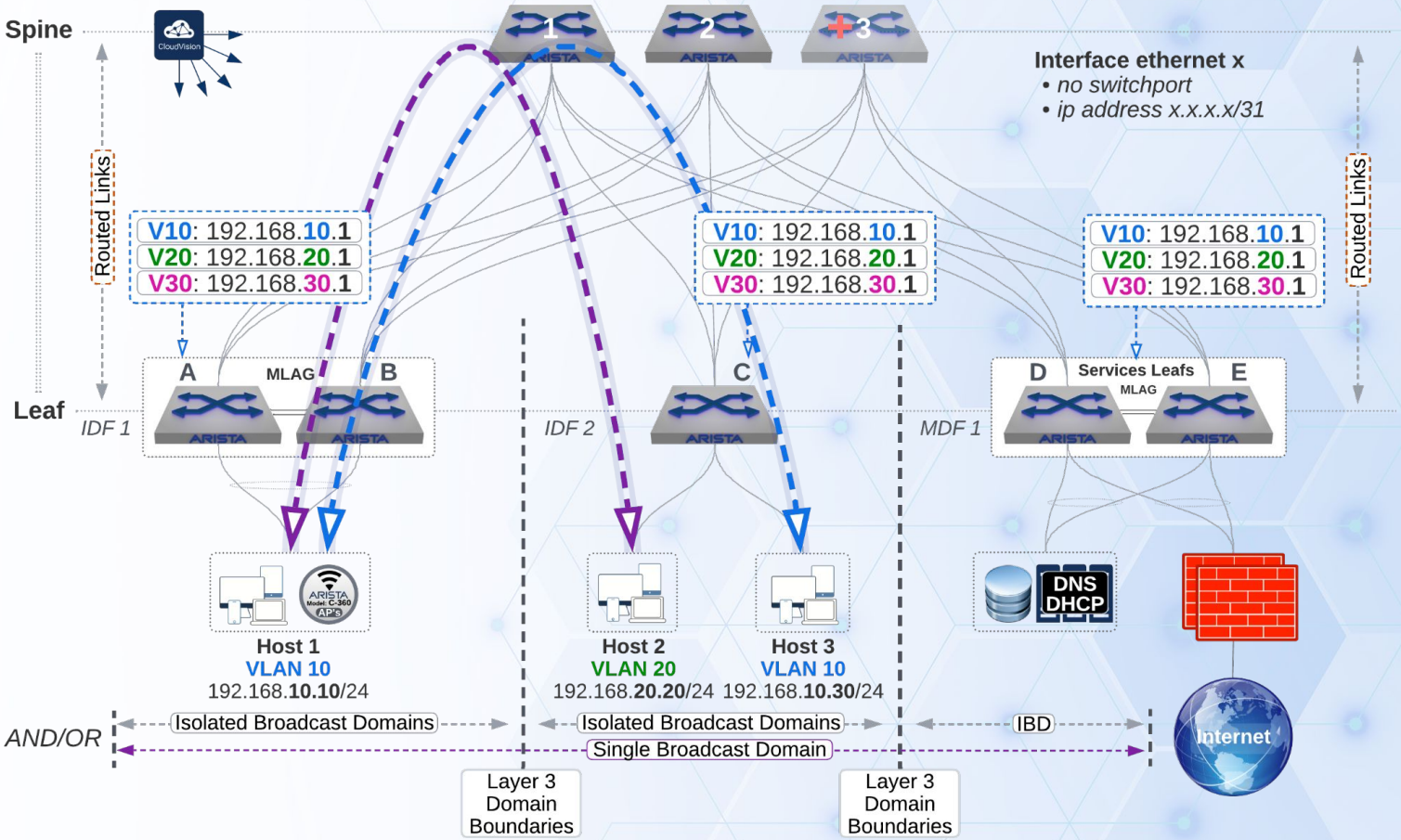


L3LS /w VXLAN (L3LS-V):

NH

Attributes

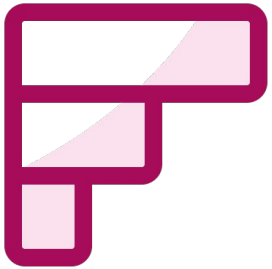
- Uses either VXLAN or MPLS as the data plane technology
- Encapsulates host-transmitted frames in the payload of an outer packet
- VXLAN enables L2 stretch
- VXLAN enables anycast gateway
- Can improve efficiency
- Native VRF support
- Introduces *many* new options



ARISTA

Review: L2LS and L3LS compared

Function	L2LS	L3LS-V
Spine-leaf links	Layer 2 switchport (vlan trunks)	Layer 3 no switchport (/31 PTP links)
Fabric Link redundancy	LACP	Equal Cost Multi Path (ECMP) routing
Hardware redundancy	MLAG (2 device redundancy)	MLAG (2 device) or ECMP (2-128 devices)
Gateway redundancy	Centralized IRB using vARP	Anycast gateway using vxlan
Physical mobility	L2 trunks broadcast everywhere	L2 stretch via vxlan
Multi-tenancy	VLAN ID only	VLAN and VRF
Multi-DC	None	VXLAN tunneling
Primary Pro / Con	Simple to configure / hard to scale	Feature rich and scalable / complex config



What kind of network design best represents the networks you have in production?

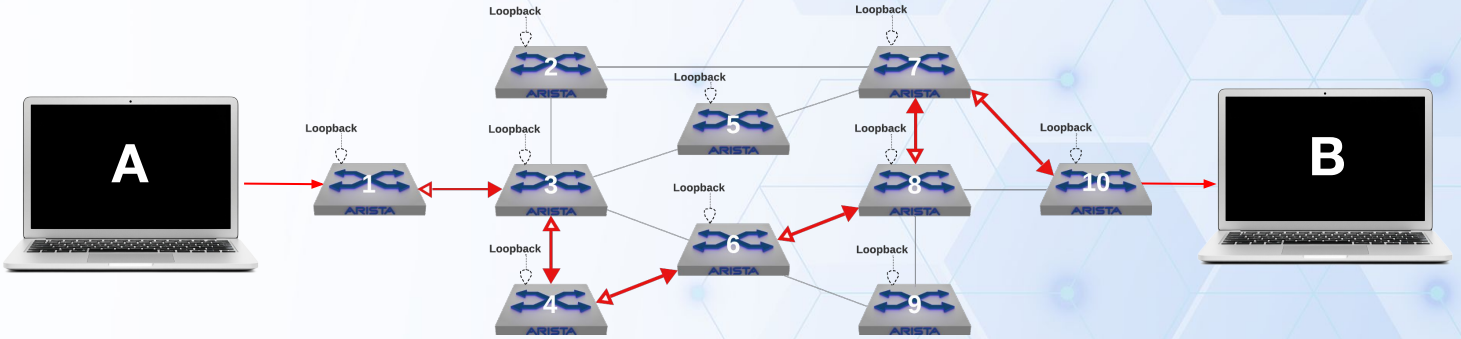
Donuts…

My rating scale: 1-10

Farmstead - **8.5**
Donutsville - **7**
Donut house - SLC - **7.5**
Donut boy - WVC - **7.5**
Darla’s Donuts - SLC - **7.5**
Duck Donuts - SJ - **7**
Dunford Donuts - West Jordan - **6.5**
Pinkbox Donuts - SG - **7**
Banbury Cross Donuts - SLC - **7**
Fresh Donuts and Deli - SLC - **8.5**
The Other Side Donuts - SLC - **7**
Judy's Donuts and Coffee - Midway - **7.5**
Parson’s Bakery - Bountiful - **8**
The Donut Run - St. George - **6.5**
Beardall's Bakery - Magna - **7.5**
Mirror Lake Highway Chevron - Kamas - **9**
Lehi Bakery - **8.5**
Provo bakery - **8.5**
Dough Miner - **7.5**
Chubby Baker - **8**
Spudly - **7**
Schmidt’s Pastry Cottage - **8**
Donut Star Cafe - **7**

Step One: The Underlay

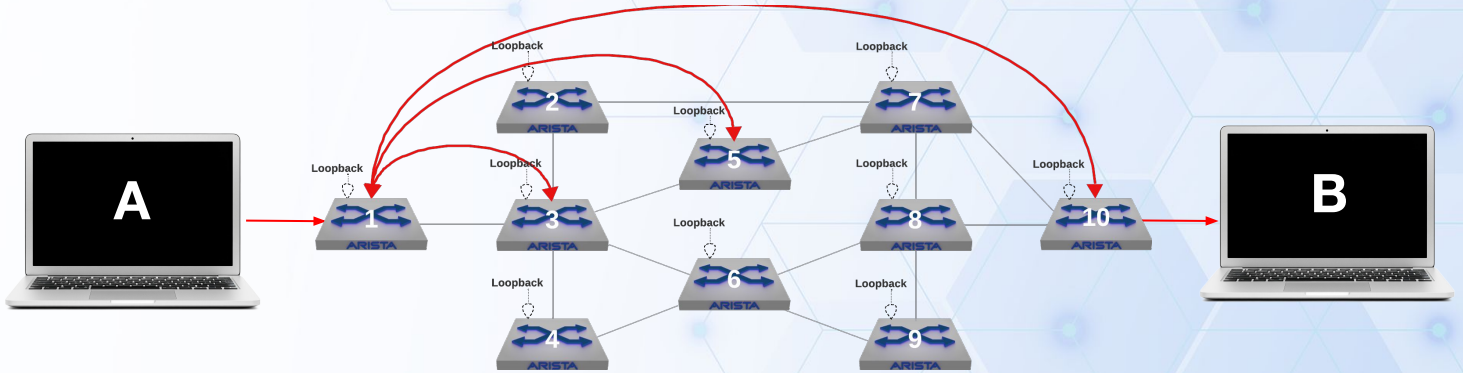
- Details:
  - It's concerned with the physical path
  - Often it is built on per-hop adjacencies
  - Path decisions are defined by routing protocol metrics
  - These per-hop adjacencies are used to advertise locally connected subnets with its neighbors
  - This includes virtual interfaces



 |

Step Two: Establish remote connectivity

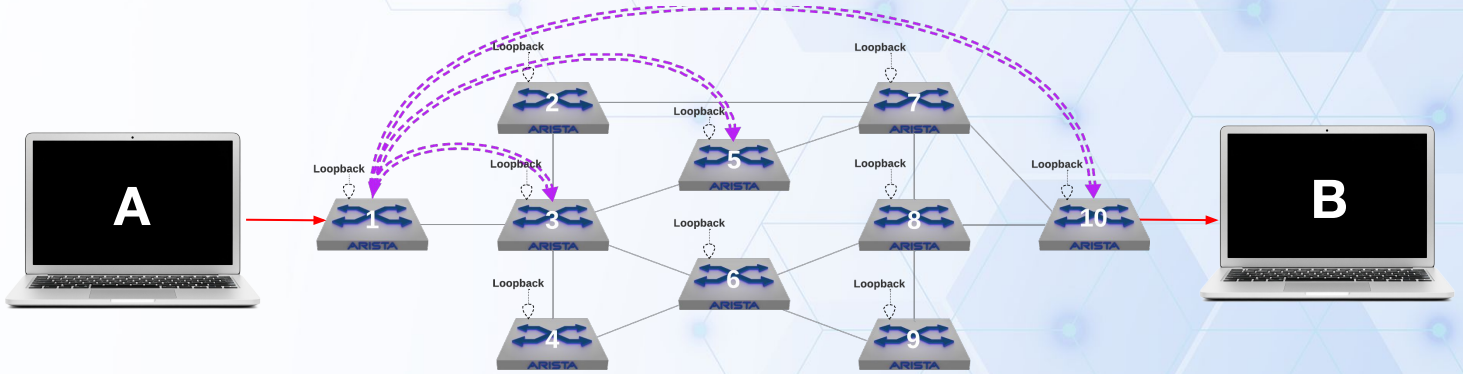
- Details:
  - Is this the Overlay?
  - What purpose does establishing remote connectivity serve?
  - Why do we advertise loopbacks as those “Anchor Points”? Why not just use IP addresses on physical interface as destinations?



 |

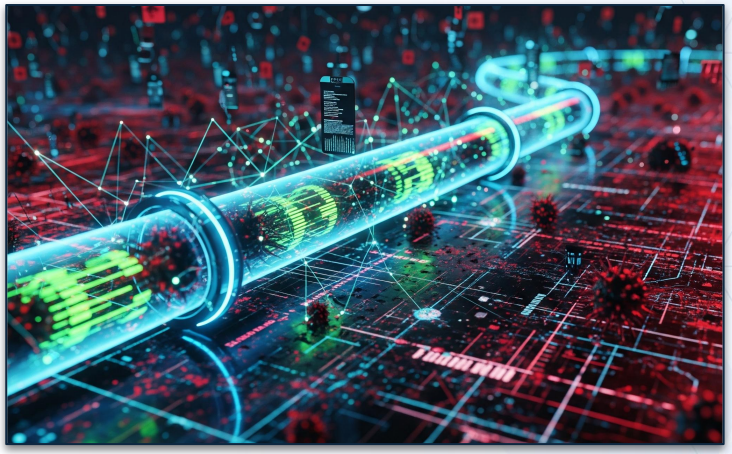
Step Three: Build VXLAN tunnels

- Details:
  - Ok, so THIS is the Overlay, right?
  - And its purpose is what, again?
  - Why can't we just use standard Underlay routing? What do we need to involve a tunnel in the first place?



Examples of an Overlay…

- Details:
  - What is the purpose of a VPN tunnel?
  - What does it protect us from?
  - How do you steer your VPN tunnel over all of the individual routers of the internet?
  - Then what is the purpose of those individual routers along the path of your VPN for?



Other examples of an Overlay

- Details:
  - The Underlay forms the foundation for the Overlay to function over.
  - The Underlay advertises loopbacks as “anchor points” for the Overlay to use
  - The Overlay uses those loopbacks build paths that defy the restrictive paths of the Underlay

What exactly is an Underlay and Overlay?

So again, why do we need tunnels?

Underlay/Overlay: Issues to solve…

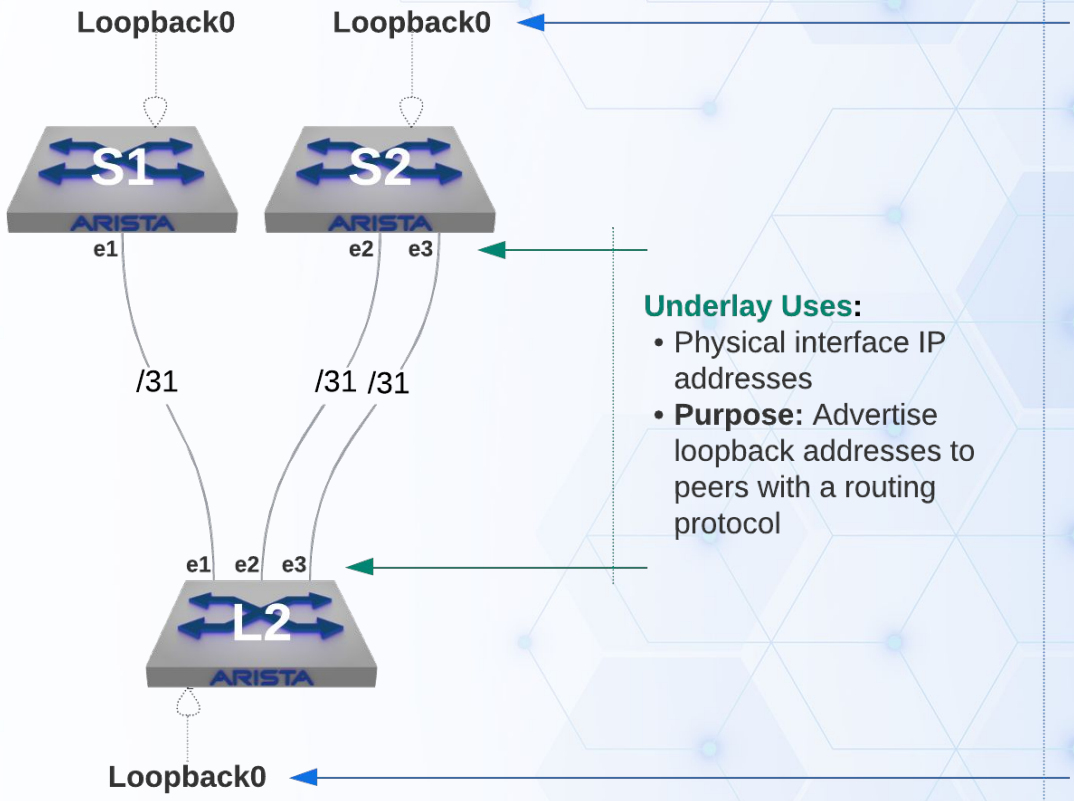
- Do what was previously impossible with other network technologies

- Problem statement 1: L2LS
  - Broadcast domain size = Failure domain size
- Problem statement 2: L2LS
  - L2LS (STP) has a diameter recommendation limit of 7 switches
- Problem statement 3: L2LS
  - Optimal traffic paths
- Problem statement 4: L3LS
  - L3LS locks prefixes to a single location in the network
- Problem statement 5: L3LS
  - VMotion a VM with L3LS
  - Wireless AP bridging - Roam between IDFs

Underlay/Overlay: Structural pieces

Attributes

- The Underlay's job is to advertise loopback addresses for the Overlay to build adjacencies on
- The Overlay's job is to add a layer of abstraction to remove architectural restrictions by enabling tunnel creation between any two switches



Underlay Uses:

- Physical interface IP addresses
- **Purpose:** Advertise loopback addresses to peers with a routing protocol

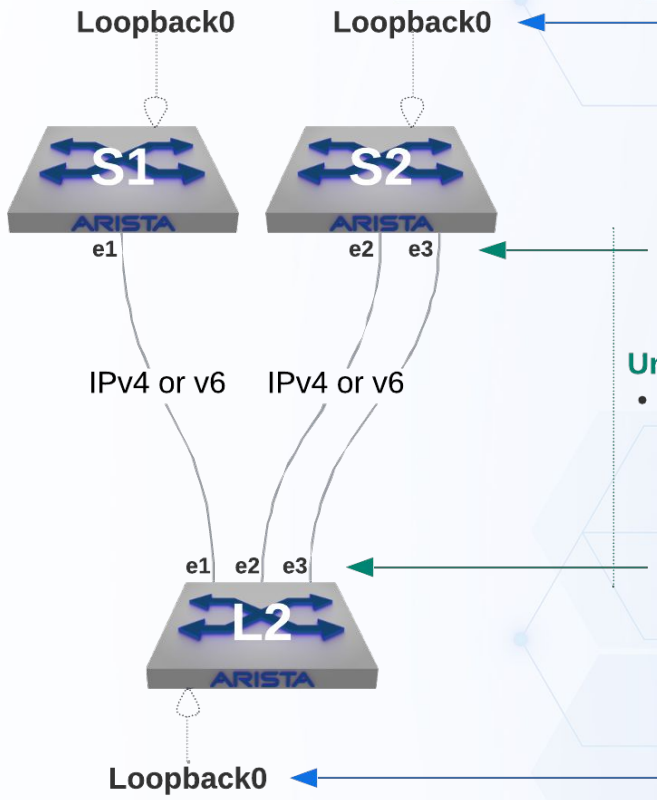
Overlay Uses:

- Virtual interface IP addresses
- **Purpose:** To form a resilient routing path based on interfaces that never go down for the Overlay to use

Underlay/Overlay: IP version options

Attributes

- Both the Underlay and the Overlay can be built using any combination of IPv4 and IPv6 address families



Underlay:
- The Underlay can use **IPv4 or IPv6 addresses**

Overlay:
- The Underlay can use **IPv4 or IPv6 addresses**

Pause and Test: L3LS-V

Try in your Lab!

Open terminal on S1-Leaf1

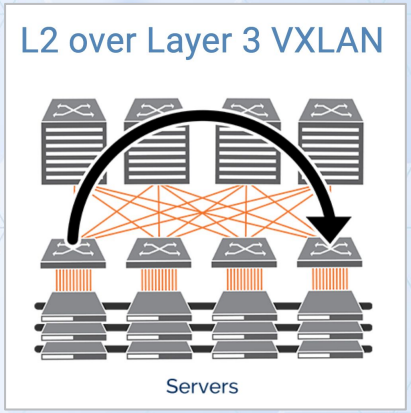
- Show run interface eth 1

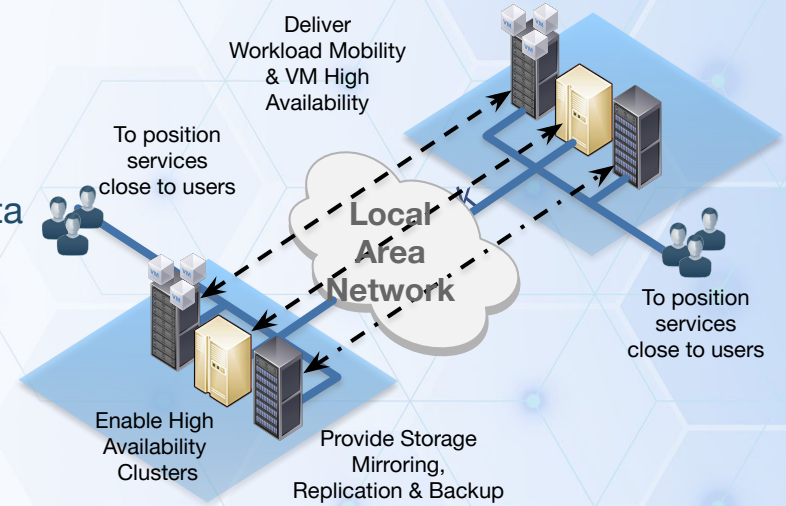
Use jumphost to change the lab to L3LS-V (option 5)

- Show run int eth 1
- Check uplinks on spine1 are now L3 routed
- Check out the int vx1 config
- Check out the int lo1 config
- ‘Show vxlan address-table’ vs ‘show mac address-table’

Virtual eXtensible LAN (VXLAN)

- RFC7348 Co-authored by Arista
- Define a MAC in IP encapsulation protocol allowing the extension of L2 domains across a L3 IP Infrastructure
- Deployed as a technology to create overlay networks across a transparent layer 3 infrastructure.
- Providing layer 2 connectivity between racks or halls of the data center, without requiring an underlying layer 2 infrastructure.
- Logical connecting geographically dispersed data centers at layer 2, as a data center Interconnect (DCI) technology.





 |

Virtual eXtensible LAN (VXLAN) - RFC

PF

Independent Submission
Request for Comments: 7348
Category: Informational
ISSN: 2070-1721

M. Mahalingam
Storvisor
D. Dutt
Cumulus Networks
K. Duda
Arista
P. Agarwal
Broadcom
L. Kreeger
Cisco
T. Sridhar
VMware
M. Bursell
Intel
C. Wright
Red Hat

August 2014

Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks

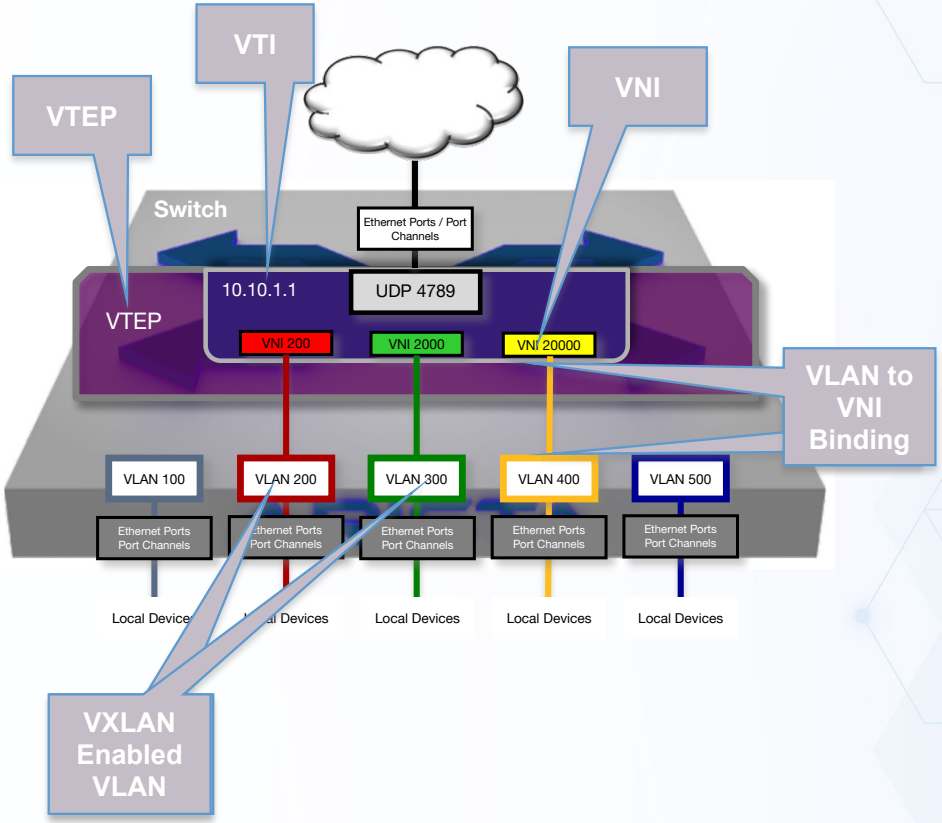
Abstract

This document describes Virtual eXtensible Local Area Network (VXLAN), which is used to address the need for overlay networks within virtualized data centers accommodating multiple tenants. The scheme and the related protocols can be used in networks for cloud service providers and enterprise data centers. This memo documents the deployed VXLAN protocol for the benefit of the Internet community.

Pause…

VXLAN Terminology: The components of the Overlay

JP



- **VTEP – VXLAN Tunnel Endpoint**
  A switch that will act as the Encapsulation/De-Encapsulation point for a VXLAN enabled VLAN

- **VTI – Virtual Tunnel Interface**
  The interface used to terminate VXLAN encapsulated traffic

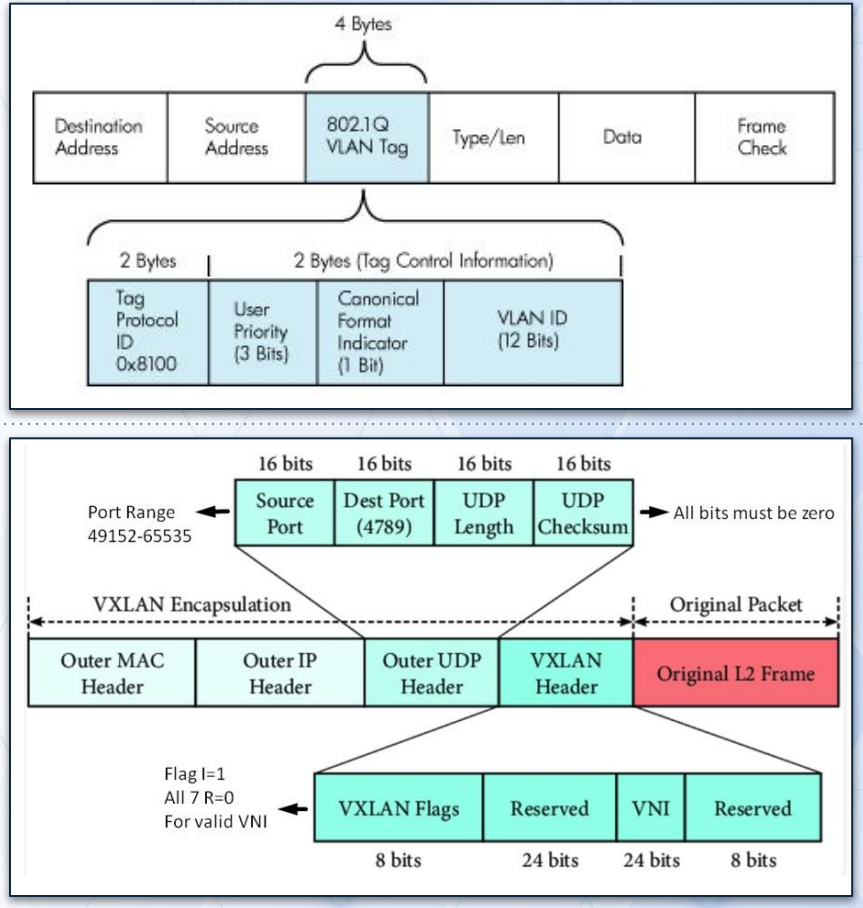
- **VNI – Virtual Network Identifier**
  VXLAN segment(s) that traverse the VXLAN

- **VXLAN Enabled VLAN**
  The VLAN that will be encapsulated into the VXLAN. VLAN ID is locally significant.

- **VLAN to VNI Bindings**
  The table that defines and binds VXLAN enabled VLANs to VNI's.

Virtual eXtensible LAN (VXLAN) - Details

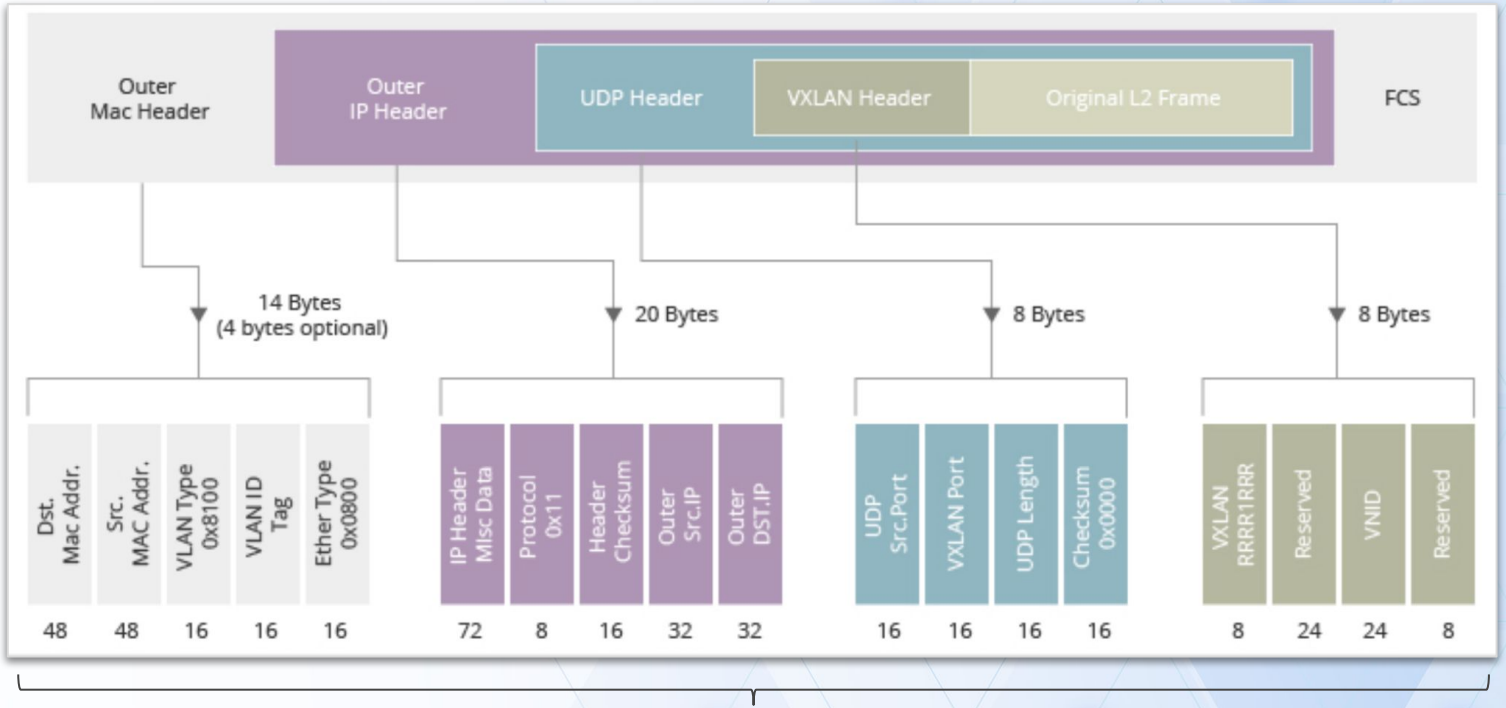
- VLAN attributes:
  - Tagged or Untagged
  - 12 bit - 4096 max
  - Carried via a 4-byte VLAN tag between the Source/Destination MAC address and Length/Type
- VNI attributes:
  - Only “Tagged”
  - 24 bit - 16,777,215 max
  - Carried within the 8-byte VXLAN header itself



ARISTA

VXLAN Fundamentals: VXLAN Packet Structure

- Frame for encapsulating L2 (Ethernet) inside L3 (IP+UDP)
- Extends 4K (12-bit) VLAN ID space into 16.7M (24-bit) VNI
- Encapsulates L2 Ethernet inside Ethernet+IP+UDP (+50 bytes overall)
- Encapsulated packet (IP) is routable and can be load balanced w/ ECMP



14+20+8+8 = 50 bytes prepended to front of existing frame
For Inner VLAN MTU 1500, IP/Underlay MTU ≥ 1550

ARISTA

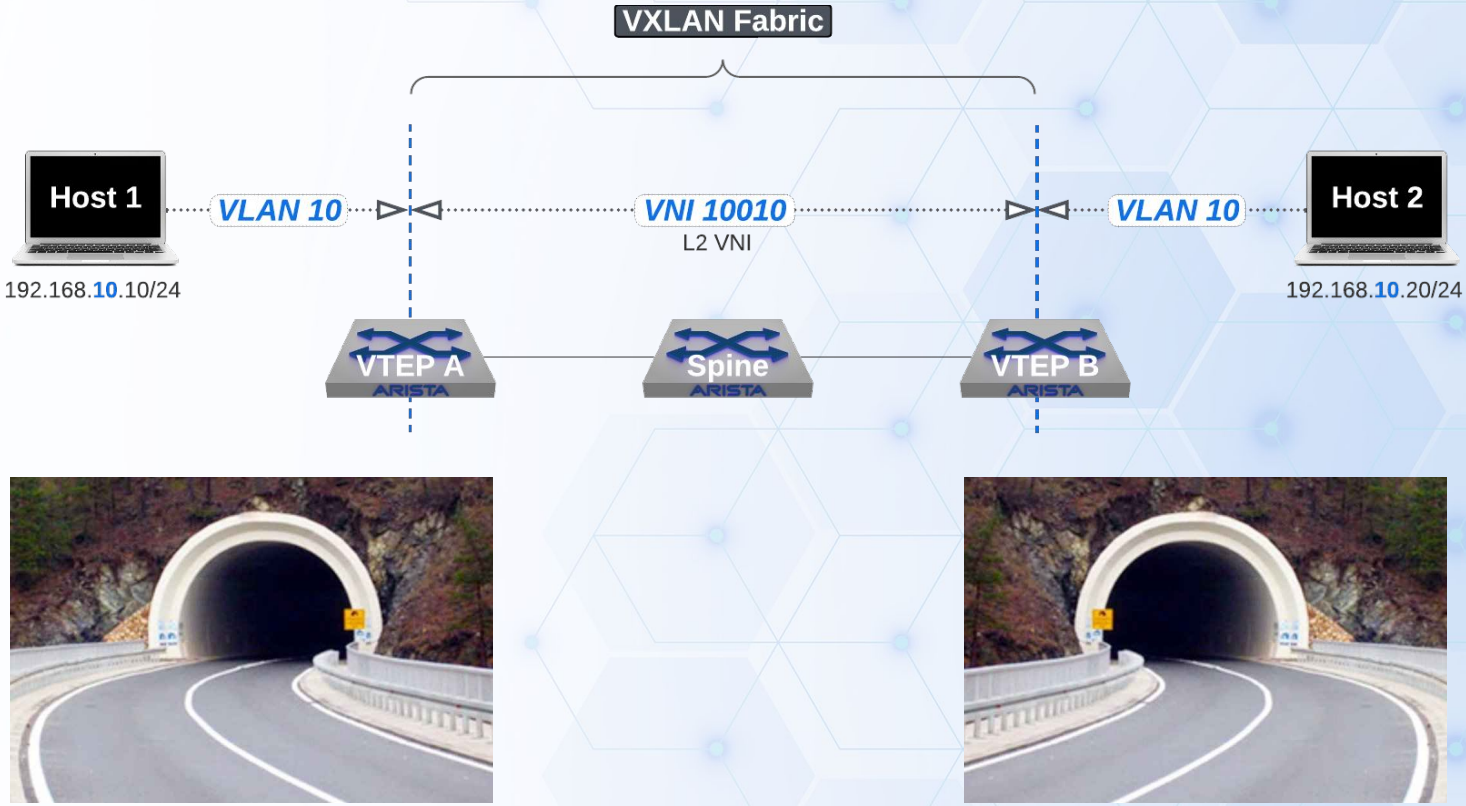
VXLAN Fundamentals: Wireshark VXLAN Packet

Prove it!

L3LS-V Zoom: L2 Stretch

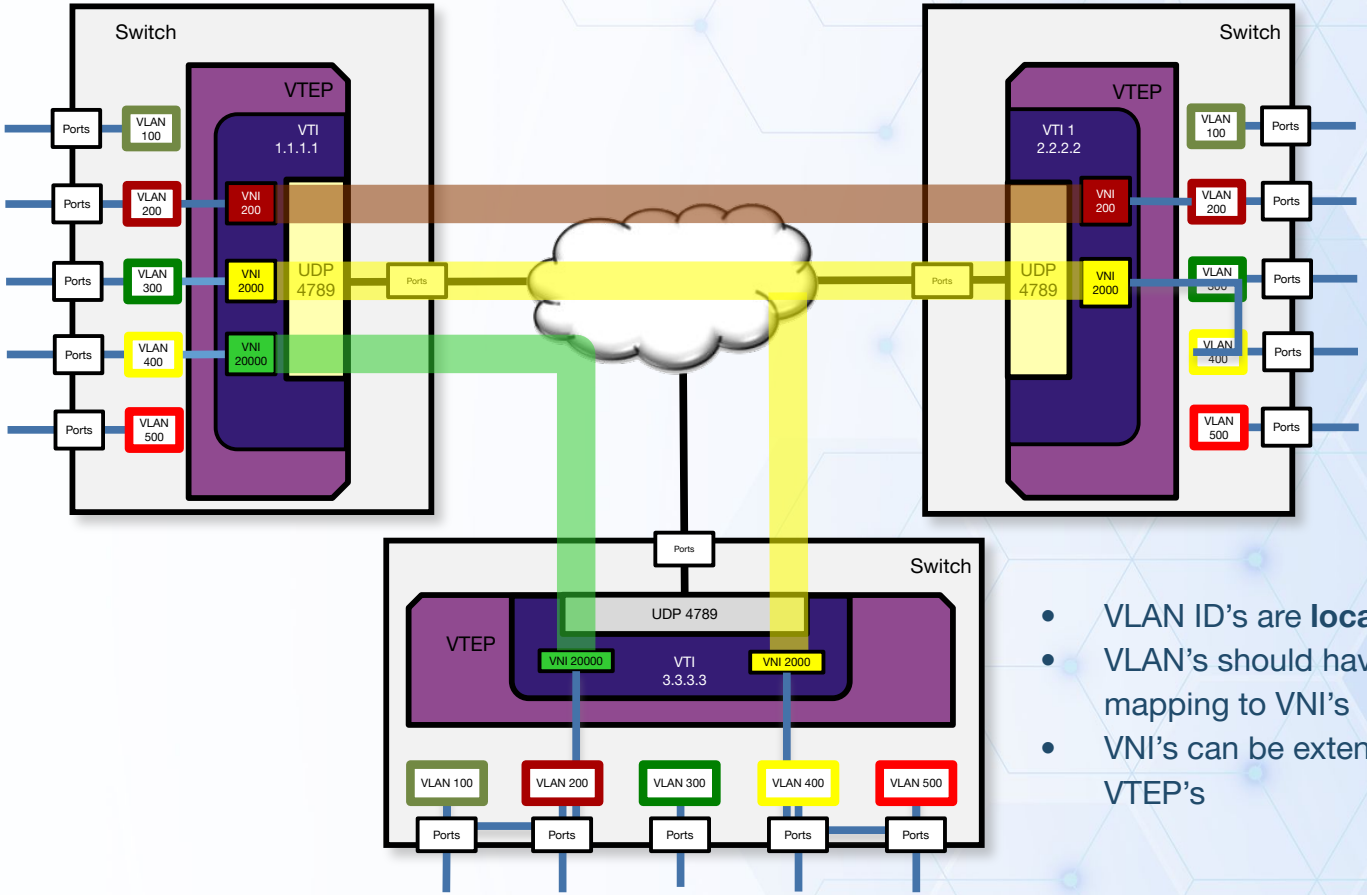
Attributes

- Stitching VLANs to VNIs
- Associate one isolation technology (VLAN) with another (VNI)



 |

VXLAN VNI's: The power of abstraction



- VLAN ID's are **locally significant**
- VLAN's should have a **one-to-one** mapping to VNI's
- VNI's can be extended to **multiple** VTEP's

 |

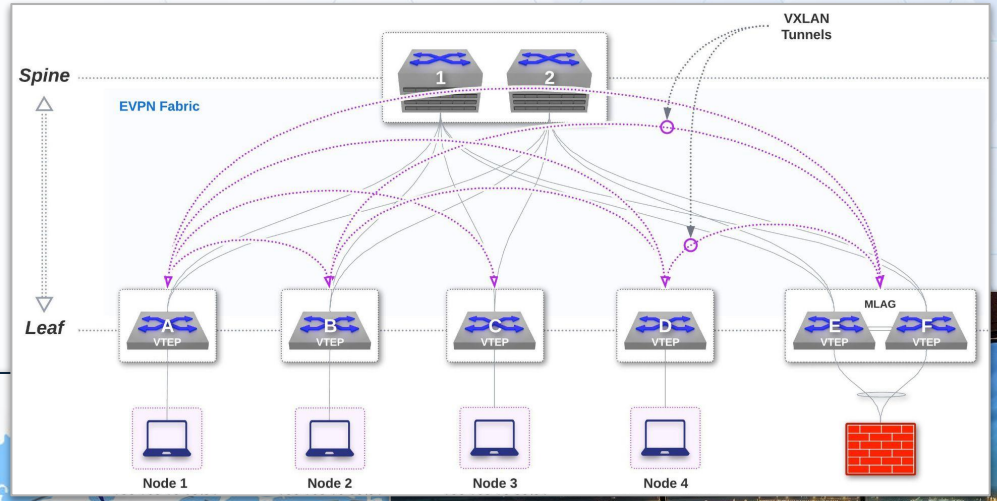
ARISTA

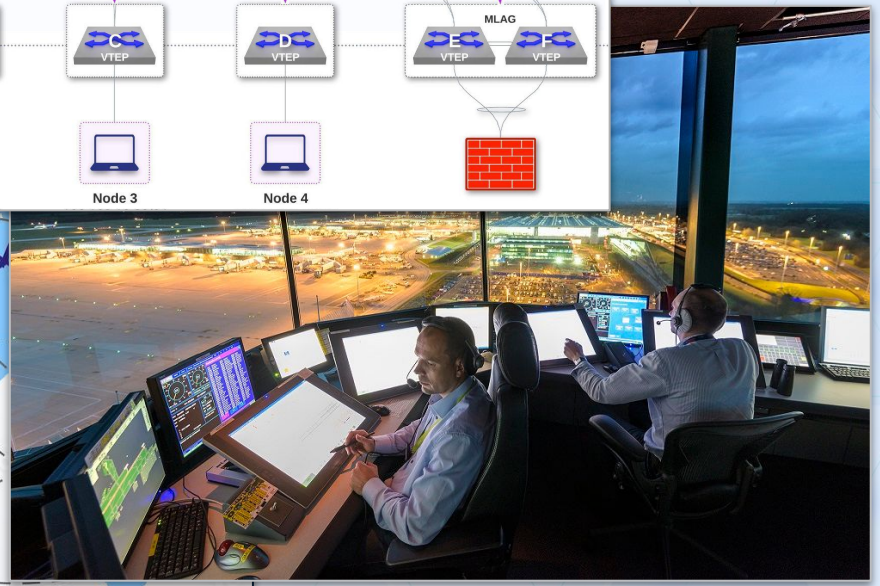
VXLAN Fundamentals: VXLAN tunnel orchestration

SD

There are two key requirements for VXLAN to work

1. **Data plane:**
   - The tunneling technology itself
   - Adds the requisite layer of abstraction
   - Easier problem to solve
2. **Control plane:**
   - How, where and when to build tunnels between two separate points?
   - More complex



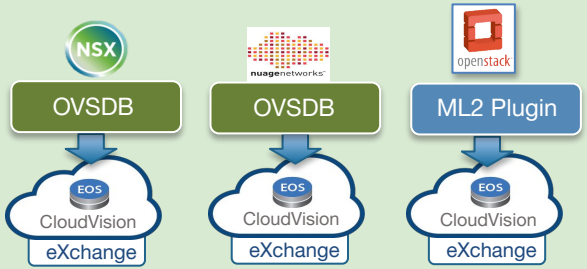


ARISTA

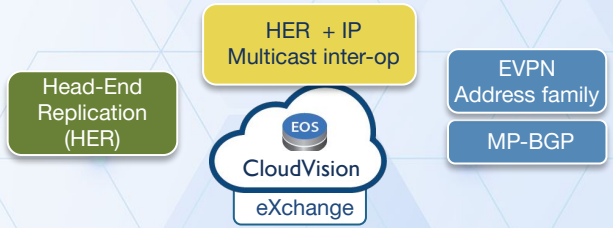
VXLAN Control-Plane Options

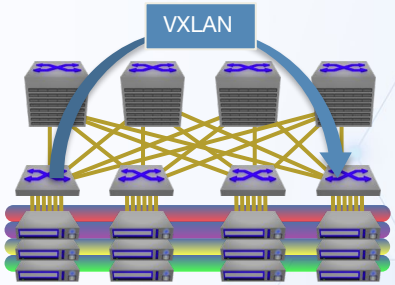
SD

Controller Model



Controller-less Model





Deprecated

HER with CloudVision eXchange (CVX)

- Local MACs and VNI binding published to CVX
- CVX dynamically distributes state to remote VTEPs
- Support for Third-party VTEP(s)
- Dynamic MAC distribution, automated flood-list provisioning
- HA Cluster support for resiliency

IP Multicast Control Plane

- VTEP join an associated IP multicast group (s) for the VNI(s)
- Unknown unicasts forwarded to VTEPs in the VNIs via IP multicast
- Support for Third-party VTEP(s)
- Flood and learn and requires IP multicast support – limited deployments

Static Flood List Head-End Replication

- BUM traffic replicated to each remote VTEPs in the VNIs
- Replication carried out on the ingress VTEP.
- Support for Third-party VTEP(s)
- MAC learning still via flood and learn but no requirement for IP multicast

The New Hotness

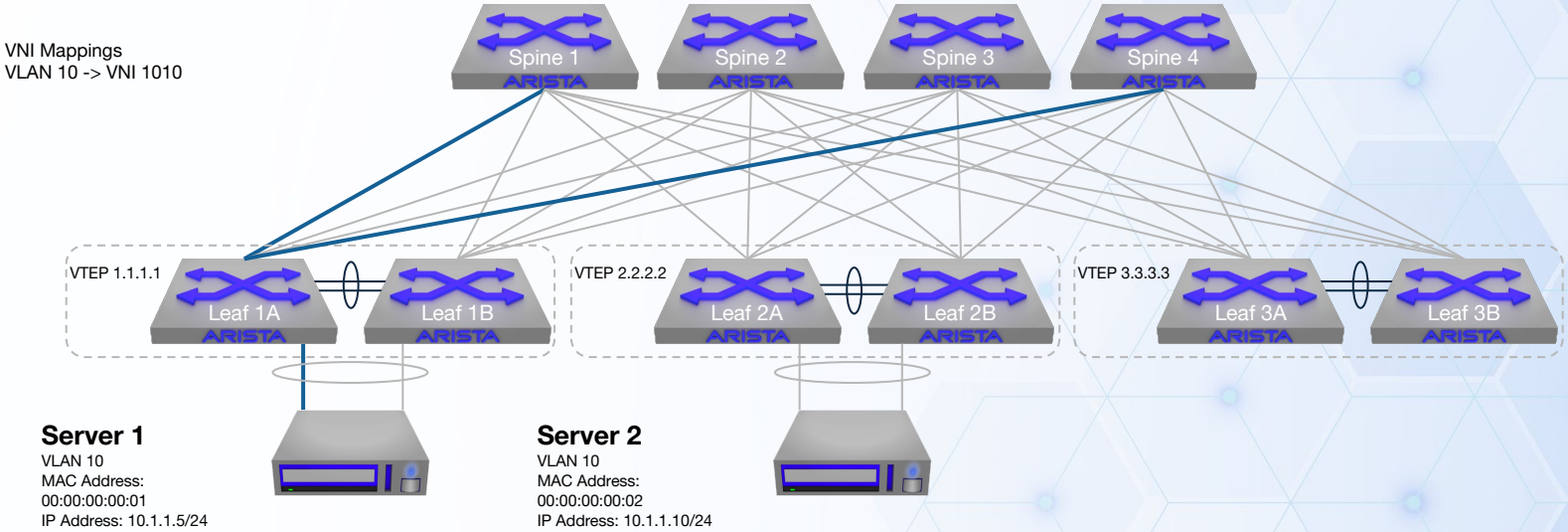
EVPN Model

- BGP to distribute local MAC/IP bindings to VTEPs
- Broadcast traffic handled via HER models
- Dynamic MAC distribution and VNI learning, via BGP intensive
- Support for Third-party VTEP(s)
- Operates outside the CVX model

Pause and Test: VXLAN /w HER

Check it out!

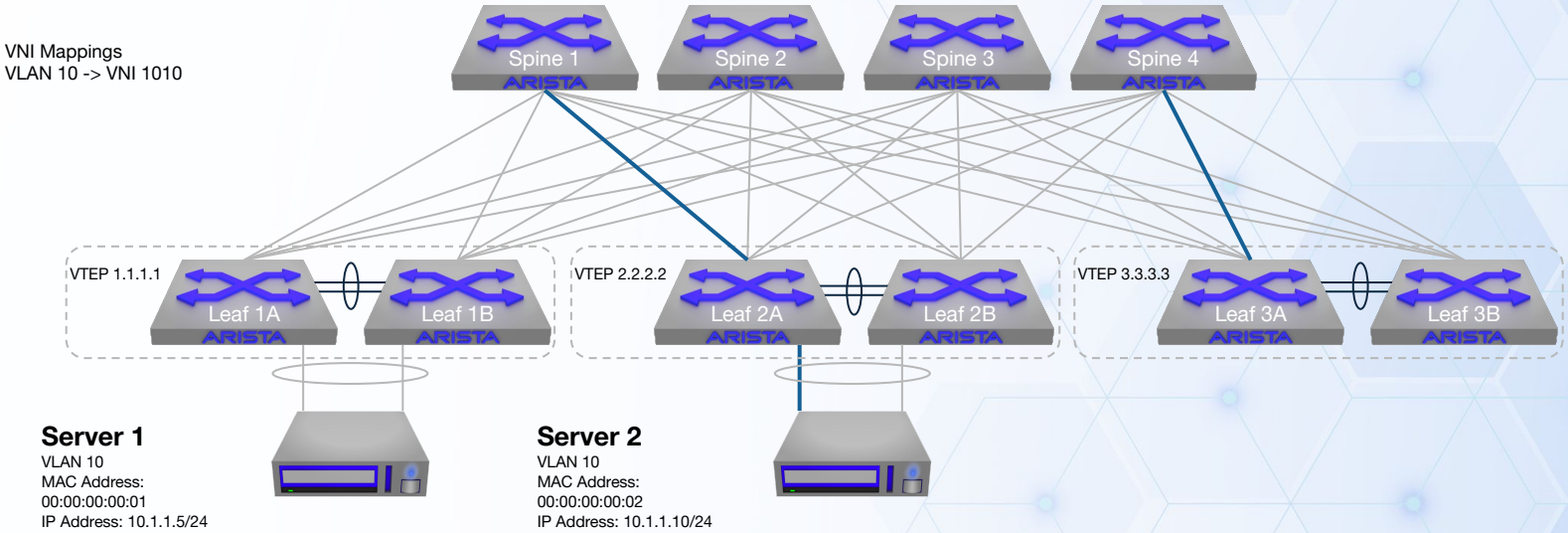
ARP Packet Walk:



ARP Request from Serv1 to Serv2

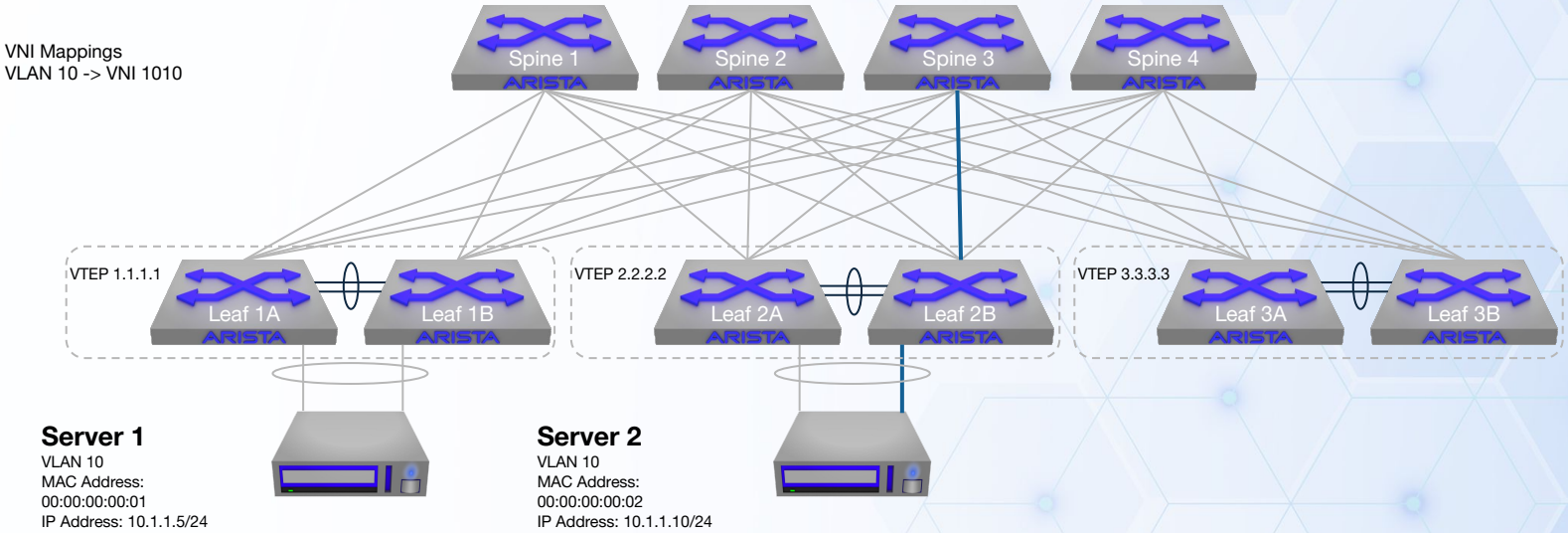
1. Server 1 sends ARP request for Server 2 over one of the LAG links (received by Leaf 1A)
2. Leaf 1A updates it's MAC table and forwards the ARP to it's MLAG peer to both synchronize the MAC table
3. Since VLAN10 is mapped to VNI1010, Leaf 1A encapsulates the ARP packet into a VXLAN packets sending to it's flood list with a source address of the local logical VTEP 1.1.1.1 and destination of the remote VTEP 2.2.2.2 and VTEP 3.3.3.3
4. Leaf 1A has four possible paths to the remote VTEP using ECMP hashing, sends the flow to one of the spine (eg sent to Spine1).

ARP Packet Walk:



5. Spine1 receives the encapsulated packet. Two paths to VTEP2 are available. Using ECMP hashing Spine 1 forwards the packet to VTEP2, it also forwards to VTEP 3 using ECMP.
6. Leaf 2A receives the encapsulated packet and de-encapsulates the VXLAN packet, learning the MAC for Sev1 as behind VTEP1.
7. Leaf 2A synchronizes it's MAC and VTEP tables with Leaf 2B
8. Since VNI1010 is mapped to VLAN10 Leaf 2A floods the ARP to all local member ports of VLAN10 and to the MLAG peer Leaf 2B.
9. Leaf 2B then floods to all local single homed VLAN 10 member ports.

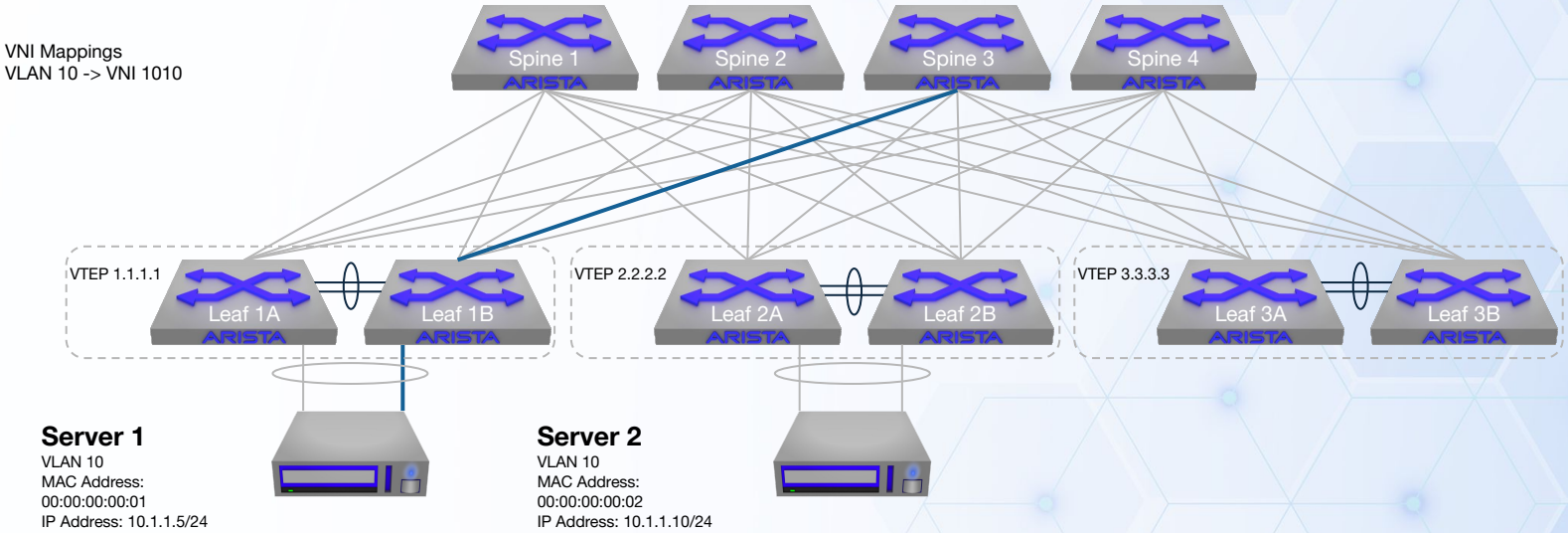
ARP Packet Walk:



Arp Response Return Path from Serv1 to Serv2

1. Serv2 sends it's ARP response back to the Serv1 MAC address via one of it's LAG links. (received by Leaf 2B)
2. Leaf 2B updates it's MAC table and synchronizes the MAC table with Leaf 2A. Since Leaf 2A and Leaf 2B know the destination MAC (Serv1) as behind VTEP1 *no flooding happens*.
3. Leaf 2B encapsulates the ARP response into a VXLAN packet with a source address of VTEP2(2.2.2.2) and Destination VTEP1(1.1.1.1)
4. Leaf 2B having four possible paths to VTEP1, performs an ECMP Hash and forward to one of the Spines (Spine 3)

ARP Packet Walk:



5. Spine 3 receives the encapsulated packet. Two paths to VTEP1 are available. Using ECMP hashing Spine 3 forwards the packet to VTEP1.
6. Leaf 1B receives the encapsulated packet and de-encapsulates the VXLAN packet it learning the MAC for Sev2 as behind VTEP2.
7. Leaf 1B synchronizes it's MAC and VTEP tables with Leaf 1A with the information of Serv2.
8. Leaf 1B forwards the de-encapsulated frame to the Port Channel from which it learnt the MAC of Serv1

VXLAN: Common Commands

VXLAN Interface Configuration Commands:

- vxlan source-interface loopback(x) - *defines the loopback interface used as the VTEP source IP for VXLAN encapsulation*
- vxlan udp-port 4789 - *sets the UDP port used for VXLAN traffic (4789 is the IANA standard)*
- vxlan vlan (xyz) vni (xyz) - *maps a VLAN to a VXLAN Network Identifier (VNI) for L2 extension across the overlay*
- vxlan flood vtep <remote vtep> - *statically defines a remote VTEP as of the flood list for a given VNI*
- vxlan virtual-router encapsulation mac-address mlag-system-id - *ensures both VTEPs in an MLAG pair use a consistent encapsulation MAC address*

Show Commands:

- show interface vxlan1- *status, source interface, encapsulation MAC*
- show vxlan vtep - *verify local and remote VTEPs*
- show vxlan flood vtep - *verify static flood list and remote VTEPs*
- show vxlan vni - *vlan-to-vni mapping validation*
- show vxlan address table - *check learned MACs per VNI*
- show vxlan counters - *monitor encapsulation/decapsulation statistics*
- show mac address-table - *confirm local MAC learning*
- show arp - *validate ARP/ND resolution*

VXLAN: Troubleshooting Commands

JH

-

VXLAN: Interface Config Commands

Command:

- vxlan source-interface Loopback(X)

This command enables MLAG peers to present themselves as a single logical VTEP. Interface Loopback(X) is configured with the same IP address on both MLAG peer members. This is commonly referred to as the “**Shared VTEP IP**” and will be used when exchanging VXLAN frames to other VTEPs in the fabric.

VXLAN: Interface Config Commands

JH

Command:

- vxlan udp-port 4789

The default VXLAN UDP port is 4789, which is used for MAC-in-UDP encapsulation of Layer 2 frames, allowing VXLAN to operate over a Layer 3 network. While 4789 is the standard, this port can be changed for security or other reasons. Keep in mind that the UDP port assignment must match across all VTEPs.

VXLAN: Interface Config Commands

Command:

- vxlan vlan (xyz) vni (xyz)

This command associates a VLAN ID with a “Virtual Network Identifier” (VNI). Packets are encapsulated with a VXLAN header that includes the VNI that is associated with the VLAN. The receiving VTEP decaps the packet and bridges the packet to the VLAN that is associated with the VNI.

*** **NOTE**: When modifying an existing VLAN to VNI ***range***, use the “**add/remove**” keyword

VXLAN: Interface Config Commands

Command:

- vxlan flood vtep <remote vtep>

This command adds a remote VTEP to the flood list for one or more VNIs, enabling Head-end replication (HER) of BUM traffic (Broadcast, Unknown unicast, Multicast) when EVPN control plane is present.

*** **NOTE**: When modifying an existing flood list, use the “**add/remove**” keyword

VXLAN: Interface Config Commands

Command:

- vxlan virtual-router encapsulation mac-address mlag-system-id

This command forces both switches in an MLAG pair to use the same MAC for VXLAN encapsulation. This ensures remote VTEPs see a single logical VTEP instead of two separate devices with potentially different encapsulation MACs. Without this command, remote VTEPs might see MAC flaps or inconsistent learning in the VXLAN address-table.

Troubleshooting VXLAN

- Why can’t my servers talk to each other? (East-West traffic)
- Users can’t access an application (North-South traffic)
- Intermittent disconnects between servers or end-hosts
- Intermittent latency between the users and applications
- MAC/ARP flapping

Troubleshooting VXLAN

Requires the same foundational skills as traditional networking—applied across both underlay & overlay.

Underlay (IP Connectivity)

- Is BGP/OSPF/IS-IS peering established and stable?
- Verify reachability: Can I ping my neighbor? Can I ping the next-hop?
- Check MTU: Is jumbo frame support consistent end-to-end?
- Validate ECMP/Load-balancing: Are all links being utilized correctly?

Routing (Underlay + Overlay)

- Do I see routes for remote VTEP loopbacks?
- Can I ping VTEP to VTEP (loopback to loopback)?
- Are flood lists configured correctly?
- Are ARP/ND entries present and correct?

L2 Overlay (MAC Learning & Flooding)

- Where did I learn this MAC? Is it local or remote? Which VTEP advertised it?
- Has the MAC moved between VTEPs or is it flapping?
- Are VNI's mapped correctly to VRF / VLAN?
- Is BUM traffic (Broadcast, Unknown Unicast, Multicast) being handled correctly?

Troubleshooting: VXLAN Bridging

Can’t ping host in the same VLAN/Subnet?

- Check MAC and VXLAN address tables.
- Check if interface “**Vx1**” is part of the vlan (**show vlan**)
- Check VLAN to VNI Mappings are consistent on all VTEPs (**show vxlan vni**)
- VXLAN tunnel(s) may not be working properly between SRC and DST VTEPs
- Check if loopback IP’s of VTEPs are present in flood list (**show int vxlan1**)
- Check routes for loopback IP reachability
- Check MTU along path

Troubleshooting: VXLAN Routing

Can’t ping host in different VLAN/Subnet?

- Traceroute to DST IP address to see which hop the packet reaches
- Check ARP/MAC tables for source/destination on local VTEPs
- Check for destination VTEP IP in the route tables
- Confirm Gateway(s) is configured properly (Anycast/Centralized)
- Check if the VMACs are configured properly (should be the same with anycast)
- Check VLAN/VNI mapping and Flood Lists
- Check MTU along the path

Troubleshooting: Intermittent Packet Loss

Intermittent ping loss from host to host?

- Ping to determine any loss patterns
- Check VTEP to VTEP (loopback to loopback) connectivity
- Check MTU along the path (test different packet sizes)
- Check logs/events for MAC/ARP flaps
- Check for L2 loops in the network - STP churn?
- Check if drops are happening fabric-wide or just to specific VTEP(s)
    - Use tcpdump to determine which device(s) is dropping the packets
- Check for errors/discards, queue counters, platform drops, etc

Troubleshooting: VXLAN MAC/ARP Flaps

MAC/ARP flapping?

- MAC flaps happening port-channel to port-channel or VXLAN int to port-channel
- Trace MAC to device it resides on (server/host,VM,VTEP)
- For ARP flaps check mappings that are changing and what device the IP belongs to
- Check for L2 loops in the network - STP churn?
- Check VMAC is configured correctly on all Routed VTEPs
- MAC/ARP aging adjustments? (Reduce flooding when MAC ages out)
- Check for flooding of BUM packets (high BW utilization on MLAG peer links and/or spine uplinks)

Troubleshooting: Missing VTI Config

Current Config:

s1-leaf1#
```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

s1-leaf2#
```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

What happens if we remove the source-interface?

s1-leaf1#
```
interface Vxlan1
   no vxlan source-interface Loopback1
```

s1-leaf2#
```
interface Vxlan1
   no vxlan source-interface Loopback1
```


Troubleshooting: Missing VTI Config

Troubleshooting Tools

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show vxlan vni` → vlan-to-vni mapping validation
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping

Troubleshooting: Missing VTI Config

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?

Checks

- What do the MAC / VXLAN address tables show?
- Look at interface vxlan1 configuration - is the interface up?
- Correct source-interface defined?

Troubleshooting: Missing VTI Config

JH

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   ...
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   ...
```

Validate things are working again

Troubleshooting: Misconfigured VTI

Current Config:

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

What happens if we change the source-interface to Loopback0?

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback0
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback0
```


Troubleshooting: Misconfigured VTI

Troubleshooting Tools

- `ping from host2 to host1` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show vxlan vni` → vlan-to-vni mapping validation
- `ping from host1 to host2` → packet loss between end-hosts?
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan flood vtep` → verify static flood list and remote VTEPs

Troubleshooting: Misconfigured VTI

What just happened??

- Did anything break?
- Why or why not?
- What issues might arise from this configuration?
- What if we shutdown one of the src host uplinks, does anything change?

Checks

- What do the MAC / VXLAN address tables show?
- Look at interface vxlan1 configuration - is the interface up?
- Correct source-interface defined?

Troubleshooting: Misconfigured VTI

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   ...
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   ...
```

Validate things are working again

Troubleshooting: Misconfigured VNI

Current Config:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

What happens if we change the vni to a different value?

```
s1-leaf1#
interface Vxlan1
   vxlan vlan 112 vni 212
```

```
s1-leaf2#
interface Vxlan1
   vxlan vlan 112 vni 212
```


Troubleshooting: Misconfigured VNI

Troubleshooting Tools

- `ping from host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan vni` → vlan-to-vni mapping validation

Troubleshooting: Misconfigured VNI

What just happened??

- Did anything break?
- Why or why not?
- What can we look at to help pinpoint the issue?

Checks

- What do the MAC / VXLAN address tables show?
- Look at interface vxlan1 configuration?
- Verify VLAN-to-VNI mappings are correct?

Troubleshooting: Misconfigured VNI

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan vlan 112 vni 112
   ...
```

```
s1-leaf2#
interface Vxlan1
   vxlan vlan 112 vni 112
   ...
```

Validate things are working again

Troubleshooting: Missing Flood List

Current Config:

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

What happens if we remove the remote VTEP from the flood list?

s1-leaf1#

```
interface Vxlan1
   no vxlan flood vtep 10.111.252.3
```

s1-leaf2#

```
interface Vxlan1
   no vxlan flood vtep 10.111.252.3
```


Troubleshooting: Missing Flood List

Troubleshooting Tools

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show vxlan vni` → vlan-to-vni mapping validation
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan flood vtep` → verify static flood list and remote VTEPs

Troubleshooting: Missing Flood List

What just happened??

- Did anything break?
- Why or why not?
- What can we look at to help pinpoint the issue?

Checks

- What do the MAC / VXLAN address tables show?
- Look at interface vxlan1 configuration - is the interface up?
- Check that the flood lists have the correct VTEPs configured.

Troubleshooting: Misconfigured Flood List

Current Config:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan flood vtep 10.111.253.3
```

What happens if we enter an incorrect IP address in the remote VTEP flood list?

```
s1-leaf1#
interface Vxlan1
   vxlan flood vtep 10.111.254.3 - IP of leaf3 Lo0
```

```
s1-leaf2#
interface Vxlan1
   vxlan flood vtep 10.111.254.4 - IP of leaf4 Lo0
```


Troubleshooting: Missing Flood List

Troubleshooting Tools

- `ping from host1 to host2` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show vxlan vni` → vlan-to-vni mapping validation
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan flood vtep` → verify static flood list and remote VTEPs

Troubleshooting: Misconfigured Flood List

JH

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan flood vtep 10.111.253.3
   ...
```

```
s1-leaf2#
interface Vxlan1
   vxlan flood vtep 10.111.253.3
   ...
```

Validate things are working again

Troubleshooting: L2 Loop

What happens if we introduce a L2 loop into the environment?

NOTE: Please don’t try this in your labs!!

Troubleshooting: Missing Flood List

Troubleshooting Tools

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show vxlan vni` → vlan-to-vni mapping validation
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan flood vtep` → verify static flood list and remote VTEPs
- `show log 10` → look at the last 10 log messages

Troubleshooting: L2 Loop

What just happened??

- Did anything break?
- Why or why not?
- What can we look at to help pinpoint the issue?

Checks

- What do the MAC / VXLAN address tables show? Any moves?
- Look at interface vxlan1 configuration - is the interface up?
- Check that the flood lists have the correct VTEPs configured.
- Check logs for MAC flapping.

Workshop Agenda Overview Day 2

10:15am - 12:15am - Recap on Day 1 - Why EVPN (again?) - Use cases

- EVPN VXLAN
  - MBGP
    - What is an address family?
    - Common EVPN Route types - 2,3,5
  - Underlay/Overlay (again) - BGP for both?
- L2EVPN & L3EVPN

12:15 pm - 1:00 pm - Lunch

- FOOD!

1:00pm - 4:00pm - Even more fun!

- Routing Models
  - External Device (Firewall, Router)
  - IRB options
    - Centralized IRB
    - Symmetric IRB
- Troubleshooting EVPN
- Labs

Welcome!

2025 Arista EVPN Workshop - Day 2

Workshop Agenda Overview Day 2

JP

10:15am - 12:15am - Recap on Day 1 - Why EVPN (again?) - Use cases

- EVPN VXLAN
  - MBGP
    - What is an address family?
    - Common EVPN Route types - 2,3,5
  - Underlay/Overlay (again) - BGP for both?
- L2EVPN & L3EVPN

12:15 pm - 1:00 pm - Lunch

- FOOD!

1:00pm - 4:00pm - Even more fun!

- Routing Models
  - External Device (Firewall, Router)
  - IRB options
    - Centralized IRB
    - Symmetric IRB
- Troubleshooting EVPN
- Labs

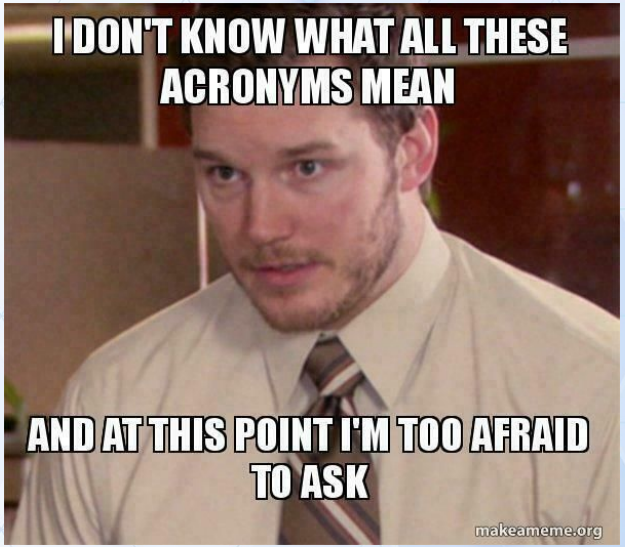
Before We Get Started

JS

1. SSH or web console to lab
2. 97. Additional Labs (labs)
3. 2. EVPN Labs (evpn-labs)
4. 4. Layer 2 and 3 EVPN VXLAN Lab (l2l3evpn) - Site 1 Only with MLAG

Recap on Day 1

- DC/Campus Architectures
- Fundamental network operation review
- MLAG (LACP) and vARP (First hop redundancy)
- L2LS vs L3LS
- Underlay and Overlay review
- Introduce VXLAN
- VXLAN Fundamentals and Operations
- Configuration Overview - vxlan1 interface
- ARP Packet Walk
    - L2LS vs. VXLAN
- Troubleshooting Basic VXLAN

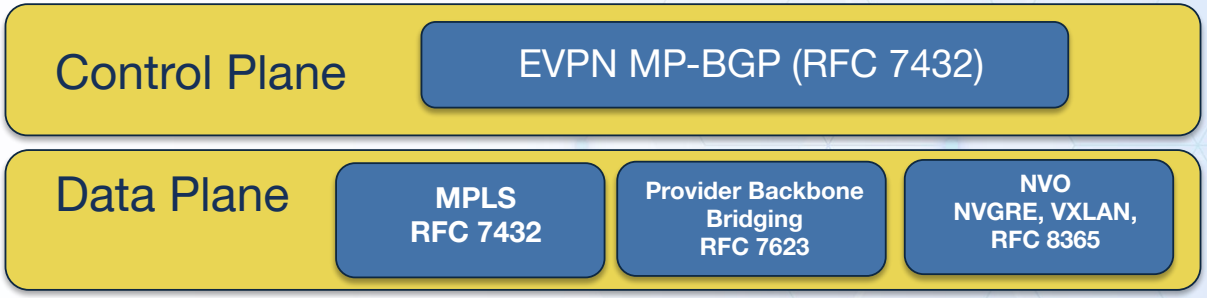


Don’t Worry

We’ll be diving into advanced networking topics.
Don’t worry if it doesn’t all click right away understanding these concepts takes time and practice.

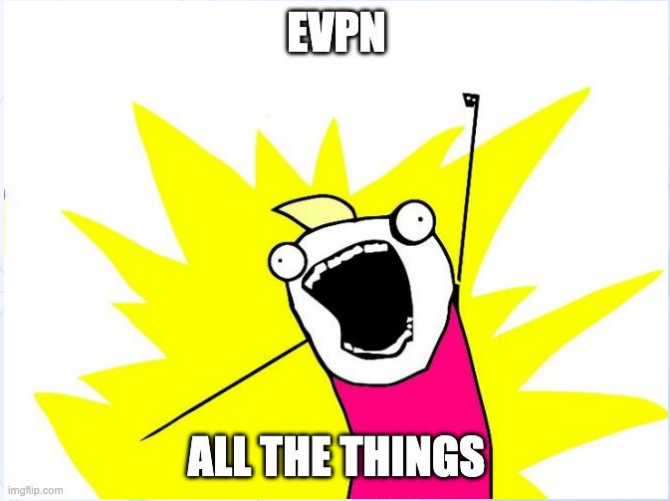
EVPN

- EVPN, IETF defined standard RFC 7432
  - RFC 7432 – MPLS forwarding plane – Metro and WAN focus
  - RFC 8365 – VXLAN, NVGRE, MPLSoGRE – Data Center focus
  - RFC 7623 – Provider Backbone Bridging – Metro Ethernet focus
- Specifics BGP control plane and new address family to advertise MAC/IP and IP prefixes.
- Providing Layer 2 and 3 VPN services on single interface, with a single MP-BGP control plane.
- Originally Introduced with MPLS in mind for the main encapsulation method
  - Designed to allow other encapsulation methods
- EVPN vs VPNv4 and MPLS L2VPN, EoMPLS, VPLS, etc…



Why EVPN (again?) - Use cases:

- Data Center
  - Scale (up and down)
  - Future Requirements
  - Ease of Segmentation
  - L2 Stretch
    - Less dependency on STP
    - Smaller STP domains
- Service Provider
  - Collapse multiple complex protocols to a single simple protocol
- Campus
  - See Data Center

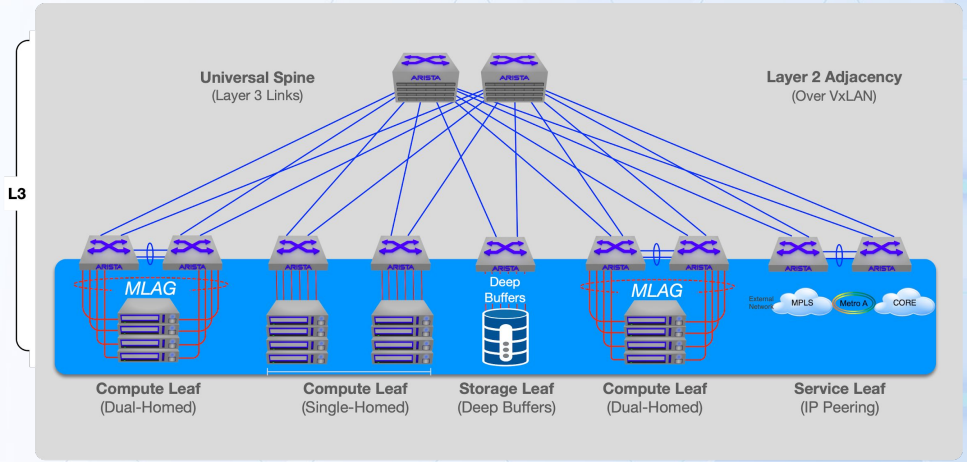


EVPN value proposition review:

1. Route table isolation using VRFs for Layer 3
    a. Similar to Broadcast domain isolation using VLANs for Layer 2
2. Proactive endpoint location advertisements
3. ARP suppression
    a. Marginalizing broadcasts
4. IP address overlap support
    a. 10.10.10.0/24
    b. 10.10.10.0/24
5. Intra/Inter-location L2/L3 extension
    a. VM services like VMWare's Vmotion can migrate a VM from one location to another without disrupting network communication
6. Host Flap detection
    a. N MAC moves (5 by default) within the damping timer window (180 seconds default)
    b. VTEP's will generate a Syslog event and stop sending or processing further updates for the T2 Route
    c. What ever VTEP that advertised the flapping MAC last is the "owner" of that blacklisted MAC
7. Multi-homing across more than 2 MLAG'ed switches
    a. EVPN Multihoming allows for a device to have up to 4 LACP enabled connections upstream that are all actively forwarding traffic

Underlay

- Layer 3 Point to Point interfaces
  - Jumbo MTU 9214
- Can you use any routing protocol (eBGP recommended, more on that later)
- Shared VTEP loopback between MLAG pairs
- Fabric unique loopback for overlay peering
- Fabric addresses don’t have to be anywhere else in the network.
- MLAG pairs peer over iBGP
- Spines are EVPN router servers



Underlay

- OSPF vs isis vs iBGP vs eBGP vs ABCD
- Single protocol vs seperate
- Arista TAC Metrics
  - Hint… eBGP

Protocol	Scale	Convergence	Operational Simplicity
OSPF	Medium	Fast	Easy
IS-IS	Large	Very Fast	Moderate
iBGP	Very Large	Moderate (needs tuning)	Complex
eBGP	Hyperscale	Very Fast (with BFD)	Simple per-link

How do you pronounce network acronyms?

MP-BGP (Multiprotocol BGP)

- RFC 2858 & 4760
- Address Family Identifier (AFI)
  - iana
    - https://www.iana.org/assignments/address-family-numbers/address-family-numbers.xhtml
- Subsequent Address Family Identifiers (SAFI)
  - iana
    - https://www.iana.org/assignments/safi-namespace/safi-namespace.xhtml
- Arista History
  - ArBgp vs gated
    - `service routing protocols model multi-agent`
      - ArBgp
    - `service routing protocols model ribd`
      - gated
    - Multi-agent (ArBgp) is default as of EOS 4.30.1
    - ribd removed EOS 4.32.0

Range	Registration Procedures
1-16383	Standards Action
16384-32767	First Come First Served

Number	Description	Reference	Registration
0	Reserved		
1	IP (IP version 4)		
2	IP6 (IP version 6)		
3	NSAP		
4	HDLC (8-bit multidrop)		
5	BBN 1822		
6	802 (includes all 802 media plus Ethernet "canonical format")		
7	E.163		
8	E.164 (SMDS, Frame Relay, ATM)		
9	F.69 (Telex)		
10	X.121 (X.25, Frame Relay)		
11	IPX		
12	Appletalk		
13	Decnet IV		
14	Banyan Vines		
15	E.164 with NSAP format subaddress	[ATM Forum UNI 3.1. October 1995.][Andy_Malis]	
16	DNS (Domain Name System)		
17	Distinguished Name	[Charles_Lynn]	
18	AS Number	[Charles_Lynn]	
19	XTP over IP version 4	[Mike_Saul]	
20	XTP over IP version 6	[Mike_Saul]	
21	XTP native mode XTP	[Mike_Saul]	
22	Fibre Channel World-Wide Port Name	[Mark_Bakke]	
23	Fibre Channel World-Wide Node Name	[Mark_Bakke]	
24	GWID	[Subra_Hegde]	
25	AFI for L2VPN information	[RFC4761][RFC6074]	
26	MPLS-TP Section Endpoint Identifier	[RFC7212]	
27	MPLS-TP LSP Endpoint Identifier	[RFC7212]	
28	MPLS-TP Pseudowire Endpoint Identifier	[RFC7212]	
29	MT IP: Multi-Topology IP version 4	[RFC7307]	
30	MT IPv6: Multi-Topology IP version 6	[RFC7307]	
31	BGP SFC	[RFC9015]	
32-16383	Unassigned		
16384	EIGRP Common Service Family	[Donnie_Savage]	2008-05-13
16385	EIGRP IPv4 Service Family	[Donnie_Savage]	2008-05-13
16386	EIGRP IPv6 Service Family	[Donnie_Savage]	2008-05-13
16387	LISP Canonical Address Format (LCAF)	[David_Meyer]	2009-11-12
16388	BGP-LS	[RFC9552]	2013-03-20
16389	48-bit MAC	[RFC7042]	2013-05-06
16390	64-bit MAC	[RFC7042]	2013-05-06
16391	OUI	[RFC7961]	2013-09-25
16392	MAC/24	[RFC7961]	2013-09-25
16393	MAC/40	[RFC7961]	2013-09-25
16394	IPv6/64	[RFC7961]	2013-09-25
16395	RBridge Port ID	[RFC7961]	2013-09-25
16396	TRILL Nickname	[RFC7455]	2014-09-02
16397	Universally Unique Identifier (UUID)	[Nischal_Sheth]	2019-11-04
16398	Routing Policy AFI	[draft-ietf-idr-rpd-02]	2020-05-12
16399	MPLS Namespaces	[draft-kaliraj-bess-bgp-sig-private-mpls-labels-03]	2021-10-19
16400-65534	Unassigned		
65535	Reserved		

How should we pronounce AFI and SAFI

Common AFI & SAFI

AFI	SAFI
1 (IPv4)	1 (Unicast)
2 (IPv6)	1 (Unicast)
1 (IPv4)	128 (MPLS-labeled VPN)
2 (IPv6)	128 (MPLS-labeled VPN)
1 (IPv4)	4 (Labeled Unicast)
2 (IPv6)	4 (Labeled Unicast)
25 (L2VPN)	70 (EVPN)

MP-BGP: BGP OPEN Packet Capture

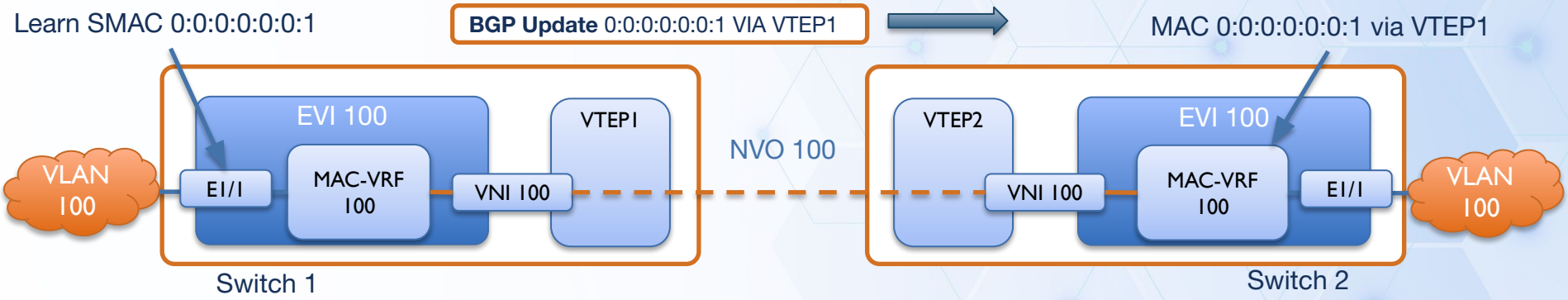
Prove it!

Routing Models in EVPN

- Centralized
  - L2 EVPN
- Asymmetric IRB
  - Anycast gateway required
  - All VLANs must be present on all switches
  - L2 EVPN
- Symmetric IRB (Integrated Routing and Bridging)
  - L3 EVPN

MAC Learning in EVPN

- MAC Learning is done via a combination of data plane (local) learning and protocol (remote) learning
- When a switch receives a L2 frame on a local interface it learns the source MAC address
- When a switch learns a MAC address on a local interface, the switch advertises the MAC address to all remote VTEPs participating in that VNI
  - Advertisements are sent using the EVPN address family of MP-BGP



 |

MAC Mobility and MAC Dampening

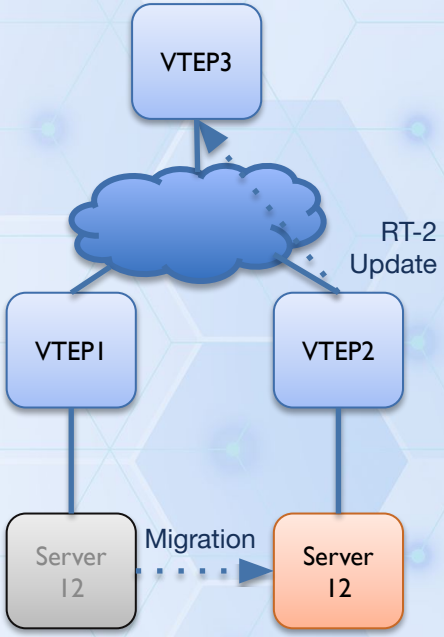
- Route updates have sequence numbers.

- If a Host moves to a new switch, it will
  - learn the MAC address of the migrated VM locally
  - Updates its MAC database and increment the sequence number for that MAC by 1.
- The new switch will send out an Type 2 update for the MAC address with the new incremental sequence number

- Remote VTEPs, including the VTEP on the original switch will
  - See the updates sequence number
  - Flush the old MAC to next hop VTEP binding
  - Populate the new MAC to next hop VTEP binding

- If the switch detects the same MAC address keeps moving
  - The switch will generate a SYSLOG message
  - The switch will stop sending and processing updates for that MAC address

MAC	Next Hop
SVR-12	VTEP1
SVR-12	VTEP2



 |

Before we keep going

Issue command on `s1-host1`

```
ping vrf 112 10.111.134.202 interval 10 repeat 10000
```


EVPN MP-BGP Route Types

Standard route types

- **Type 2: MAC Advertisement Routes**
  - **Carry MAC addresses and MAC + IP bindings (ARP entries) in a L3 EVPN**
  - **Only Carry MAC addresses in a L2 EVPN**
- **Type 5: IP Prefix Route**
  - **Host routes**
  - **Carry IP subnets**
- **Type 3: Inclusive Multicast Ethernet Tag (IMET)**
  - **Advertises VNI to VTEP mappings**
  - **Used to facilitate BUM traffic in EVPN**
  - **Type 3 routes are used to discover remote VTEPs and the VNIs they are a member of**
    - **Used to update the local flood lists**

Sneak Peek for Advance EVPN Workshop…

- Type 1: Ethernet A-D Routes (A-D = Auto Discovery)
  - Are used to announce the reachability of multi-homed Ethernet segments.
- Type 4: Ethernet Segment Routes
  - Used in multi-homed topologies to discover remote VTEPs on the same shared Ethernet segment and elect a DR.
- Types 6-8
  - Multicast
- Types 9-11 (Currently not implemented)
  - BUM enhancements

Type 3 Route advertisements and BUM Traffic – imet

- Type 3 routes are also referred to as IMET (Inclusive Multicast Ethernet Tag)
- When a new VNI is defined on a switch
  - The local VTEP will send out a Type 3 route advertisement
  - This advertisement will include all the VNIs the VTEP is participating in.
  - This allows remote VTEPs to dynamically learn about the new VTEP and the L2 segments they belong to
- Type 3 updates allow VTEPs to dynamically join or leave a L2 segment.
- When a VTEP receives a Type 3 route update, it adds the remote VTEP + VNIs to its flood list.

EVPN Route Types: imet Packet Capture & CLI

Prove it!

Pause and Test: imet

Try in your Lab!

- Show imet Spine vs Leaf

```
show bgp evpn route-type imet
show bgp evpn route-type imet 10.111.253.1 detail
```

- Where is this route coming from?
  - Spine vs Leaf
- How many routes are there?
  - Spine vs Leaf
- What is the next hop of this route?

Route Type 3 imet (non VLAN-Aware Bundle) - Spine

```
BGP routing table entry for imet 10.111.253.1, Route Distinguisher: 10.111.254.1:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1
BGP routing table entry for imet 10.111.253.1, Route Distinguisher: 10.111.254.1:134
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:134:134 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
```


Route Type 3 imet (non VLAN-Aware Bundle) - Leaf

```
BGP routing table entry for imet 10.111.253.1, Route Distinguisher: 10.111.254.2:134
Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community: Route-Target-AS:134:134 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:134:134 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
…
BGP routing table entry for imet 10.111.253.1, Route Distinguisher: 10.111.254.2:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1
```


Route Type 3 imet (VLAN-Aware Bundle) - Spine

```
BGP routing table entry for imet 112 10.111.253.1, Route Distinguisher: 10.111.254.1:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:1:1 TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1

…
BGP routing table entry for imet 134 10.111.253.1, Route Distinguisher: 10.111.254.1:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:1:1 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
```


Route Type 3 imet (VLAN-Aware Bundle) - Leaf

```
BGP routing table entry for  imet 112  10.111.253.1, Route Distinguisher:  10.111.254.1:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community:  Route-Target-AS:1:1  TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:1:1 TunnelEncap:tunnelTypeVxlan
      VNI: 112
      PMSI Tunnel: Ingress Replication, MPLS Label: 112, Leaf Information Required: false, Tunnel ID: 10.111.253.1
BGP routing table entry for  imet 134  10.111.253.1, Route Distinguisher:  10.111.254.1:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community: Route-Target-AS:1:1 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:1:1 TunnelEncap:tunnelTypeVxlan
      VNI: 134
      PMSI Tunnel: Ingress Replication, MPLS Label: 134, Leaf Information Required: false, Tunnel ID: 10.111.253.1
```


Configuration Overview: imet


```
router bgp 65101
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   !
   vlan 134
      rd auto
      route-target both 134:134
      redistribute learned
```

```
interface Vxlan1
   …
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
```


Type 2 route advertisements - mac-ip

- Type 2 routes are used to advertise MAC addresses
  - Every time a switch learns a new MAC address it sends a Type 2 update
  - This update is sent to every VTEP participating in the VNI

- Type 2 routes contain an optional IP address field
  - Can be used to carry the IP address associated with a MAC address
  - In other words the IP of the host that owns the MAC address

- When an VTEP receives a RT-2 update it also imports the remote VTEP and VNI information along with the MAC and IP information
  - This allows the MAC-VRF to maintain the VXLAN information related to a remote MAC
    - 00:00:00:00:00:01 via VTEP 3 VNI 100

- The receiving VTEP can also use the MAC + IP fields to build an ARP entry
  - This allows ARP suppression to be turned on in the IP network

EVPN Route Types: mac-ip Packet Capture & CLI

JS

Prove it!

Pause and Test: mac-ip

JS

Try in your Lab!

- Show imet Spine vs Leaf

```
show bgp evpn route-type mac-ip
show bgp evpn route-type mac-ip 001c.73c0.c616 detail
```

- Where is this route coming from?
  - Spine vs Leaf
- What is the next hop of this route?

Route Type 2 mac-ip (MAC Only) - Spine

```
BGP routing table entry for mac-ip001c.73c0.c616, Route Distinguisher: 10.111.254.1:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112 ESI: 0000:0000:0000:0000:0000
BGP routing table entry for mac-ip 001c.73c0.c616, Route Distinguisher: 10.111.254.2:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.2 (10.111.254.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112 ESI: 0000:0000:0000:0000:0000
```


Route Type 2 mac-ip (MAC Only) - Leaf

```
BGP routing table entry for mac-ip001c.73c0.c616, Route Distinguisher: 10.111.254.1:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP
contributor
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112 ESI: 0000:0000:0000:0000:0000
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:112:112 TunnelEncap:tunnelTypeVxlan
      VNI: 112 ESI: 0000:0000:0000:0000:0000
```


Route Type 2 mac-ip (MAC + IP) - Spine

```
BGP routing table entry for mac-ip001c.73c0.c616 10.111.112.201, Route Distinguisher: 10.111.254.1:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.1 (10.111.254.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:112:112 Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan
EvpnRouterMac:02:1c:73:c0:c6:12
      VNI: 112 L3 VNI: 5001 ESI: 0000:0000:0000:0000:0000
BGP routing table entry for mac-ip 001c.73c0.c616 10.111.112.201, Route Distinguisher: 10.111.254.2:112
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.2 (10.111.254.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:112:112 Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan
EvpnRouterMac:02:1c:73:c0:c6:12
      VNI: 112 L3 VNI: 5001 ESI: 0000:0000:0000:0000:0000
```


Route Type 2 mac-ip (MAC + IP) - Leaf

```
BGP routing table entry for mac-ip 001c.73c0.c616 10.111.112.201, Route Distinguisher: 10.111.254.1:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community: Route-Target-AS:112:112 Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan
EvpnRouterMac:02:1c:73:c0:c6:12
      VNI: 112 L3 VNI: 5001 ESI: 0000:0000:0000:0000:0000
  65100 65101
    10.111.253.1 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:112:112 Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan
EvpnRouterMac:02:1c:73:c0:c6:12
      VNI: 112 L3 VNI: 5001 ESI: 0000:0000:0000:0000:0000
BGP routing table entry for mac-ip 001c.73c0.c616 10.111.112.201, Route Distinguisher: 10.111.254.2:112
 Paths: 2 available
  65100 65101
    10.111.253.1 from 10.111.0.1 (10.111.0.1)
      …


#show ip route
...
 B E      10.111.112.201/32 [200/0]
           via VTEP 10.111.253.1 VNI 5001 router-mac 02:1c:73:c0:c6:12 local-interface Vxlan1
```


Configuration Overview: mac-ip (single tag)

JS

```
router bgp 65101
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   !
   vlan 134
      rd auto
      route-target both 134:134
      redistribute learned
```

```
interface Vxlan1
   …
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
```


Config: mac-ip (dual tag)


```
router bgp 65101
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   !
   vlan 134
      rd auto
      route-target both 134:134
      redistribute learned
!
   vrf TENANT
      route-target import evpn 5001:5001
      route-target export evpn 5001:5001
      redistribute connected
```

```
interface Vxlan1
   ...
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
```


Type 5 route advertisements - ip-prefix

- Type 5 routes carries IP Prefixes
  - SVI networks
  - 0.0.0.0/0

EVPN Route Types: ip-prefix Packet Capture & CLI

JS

Prove it!

 |

Pause and Test: ip-prefix

Try in your Lab!

- Show imet Spine vs Leaf

```
show bgp evpn route-type ip-prefix ipv4
```

- Where is this route coming from?
  - Spine vs Leaf
- What is the next hop of this route?
- RD

Route Type 5 ip-prefix - Spine

```
BGP routing table entry for ip-prefix 10.111.112.0/24, Route Distinguisher: 10.111.254.3:1
 Paths: 2 available
  65100 65102
    10.111.253.3 from 10.111.0.2 (10.111.0.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP head, ECMP, best, ECMP contributor
      Extended Community: Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan EvpnRouterMac:02:1c:73:c0:c6:14
      VNI: 5001
  65100 65102
    10.111.253.3 from 10.111.0.1 (10.111.0.1)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, ECMP, ECMP contributor
      Extended Community: Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan EvpnRouterMac:02:1c:73:c0:c6:14
      VNI: 5001
```


Route Type 5 ip-prefix - Leaf

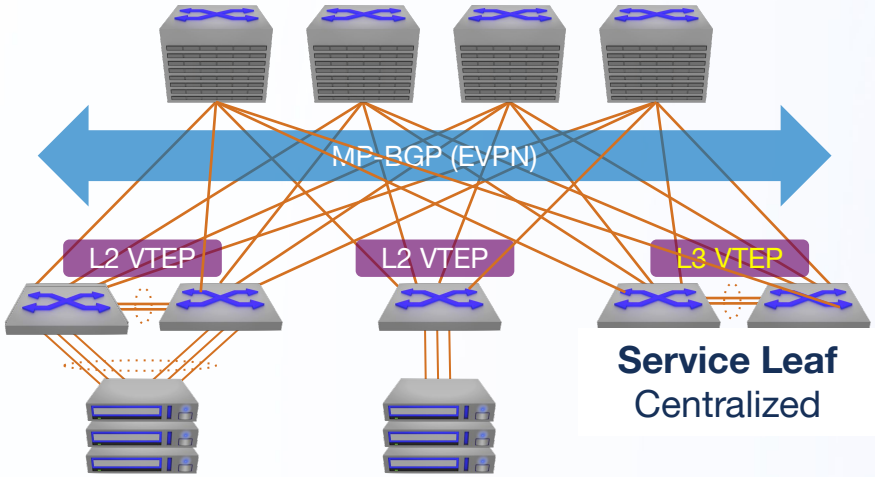
```
BGP routing table entry for ip-prefix 10.111.112.0/24, Route Distinguisher: 10.111.254.2:1
 Paths: 1 available
  65101
    10.111.253.1 from 10.111.254.2 (10.111.254.2)
      Origin IGP, metric -, localpref 100, weight 0, tag 0, valid, external, best
      Extended Community: Route-Target-AS:5001:5001 TunnelEncap:tunnelTypeVxlan EvpnRouterMac:02:1c:73:c0:c6:12
      VNI: 5001
```


Configuration Overview: ip-prefex


```
interface Vxlan1
   …
  vxlan virtual-router encapsulation mac-address mlag-system-id
  vxlan vrf TENANT vni 5001
!
router bgp 65101
   vrf TENANT
      route-target import evpn 5001:5001
      route-target export evpn 5001:5001
      redistribute connected
```

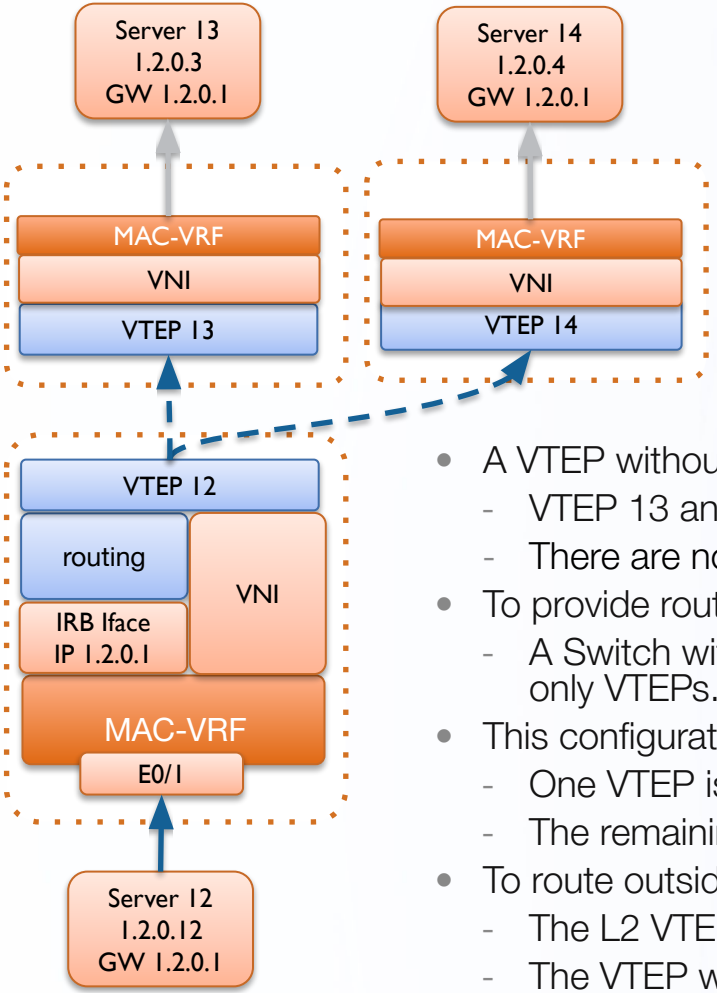
 |

EVPN Centralized Routing



- Routing occurs on Service Leafs
- Other leafs are VXLAN bridging only
- SVIs configured on L3 VTEPs or other L3 devices (Firewalls, router on a stick, etc)
- L3 VTEPs learn remote MACs via EVPN
- L3 VTEPs learn ARP via data plane (ARP bindings not known by L2 VTEPs)

L2 only VTEPs and Centralized Gateways

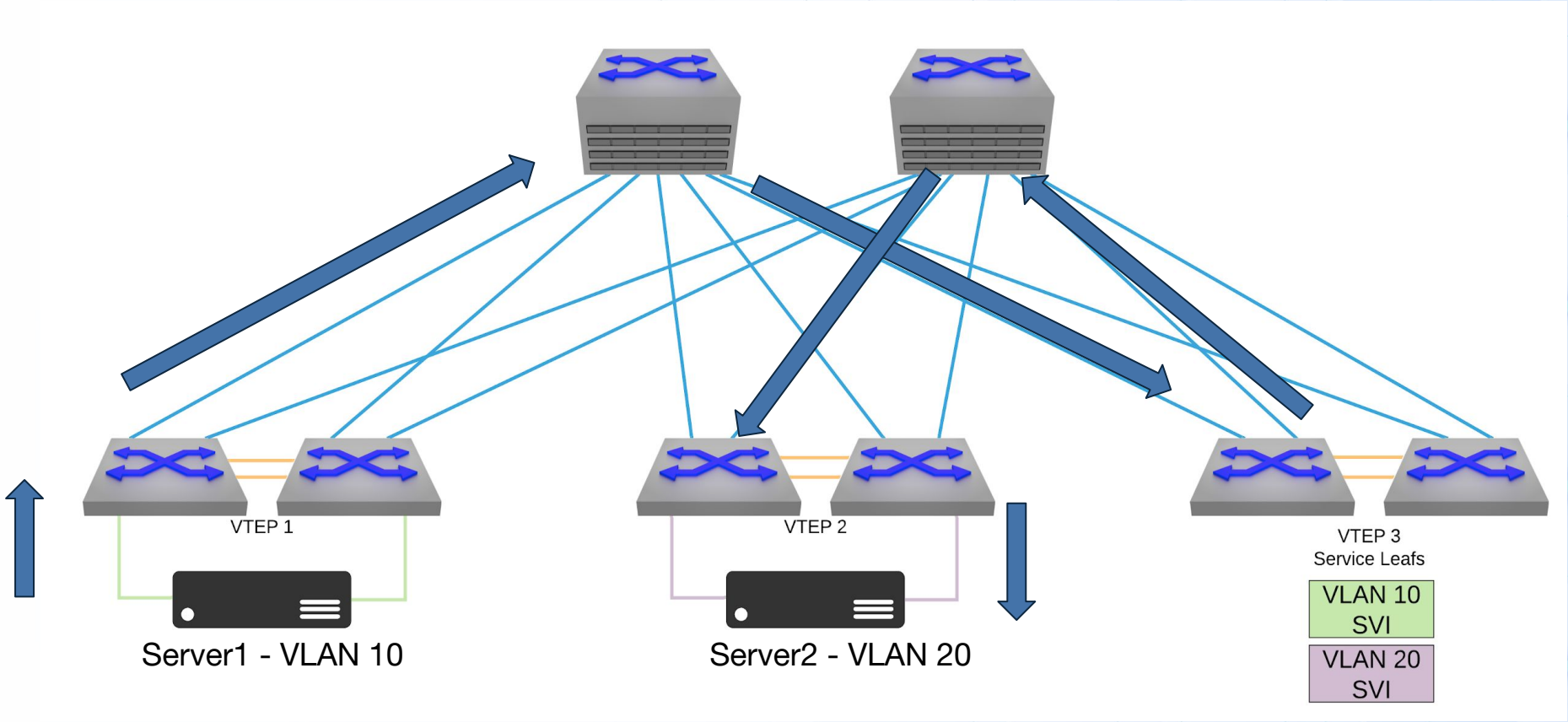


VTEP 13 and VTEP 14 are L2 only VTEPs.

- A VTEP without an IRB interface is known as a L2 only VTEP
  - VTEP 13 and VTEP 14 are L2 only VTEPs
  - There are no IRB interface, therefore there is no support for IP routing
- To provide routing services to a L2 only VTEP
  - A Switch with an IRB interface can share its IRB interface with multiple remote L2 only VTEPs.
- This configuration is known as a centralized gateway
  - One VTEP is configured with an IRB interface
  - The remaining VTEPs do not have IRB interfaces.
- To route outside the L2 segment
  - The L2 VTEPs VXLAN bridge to the VTEP with the IRB interface
  - The VTEP with the IRB interface routes packet via the IRB interface

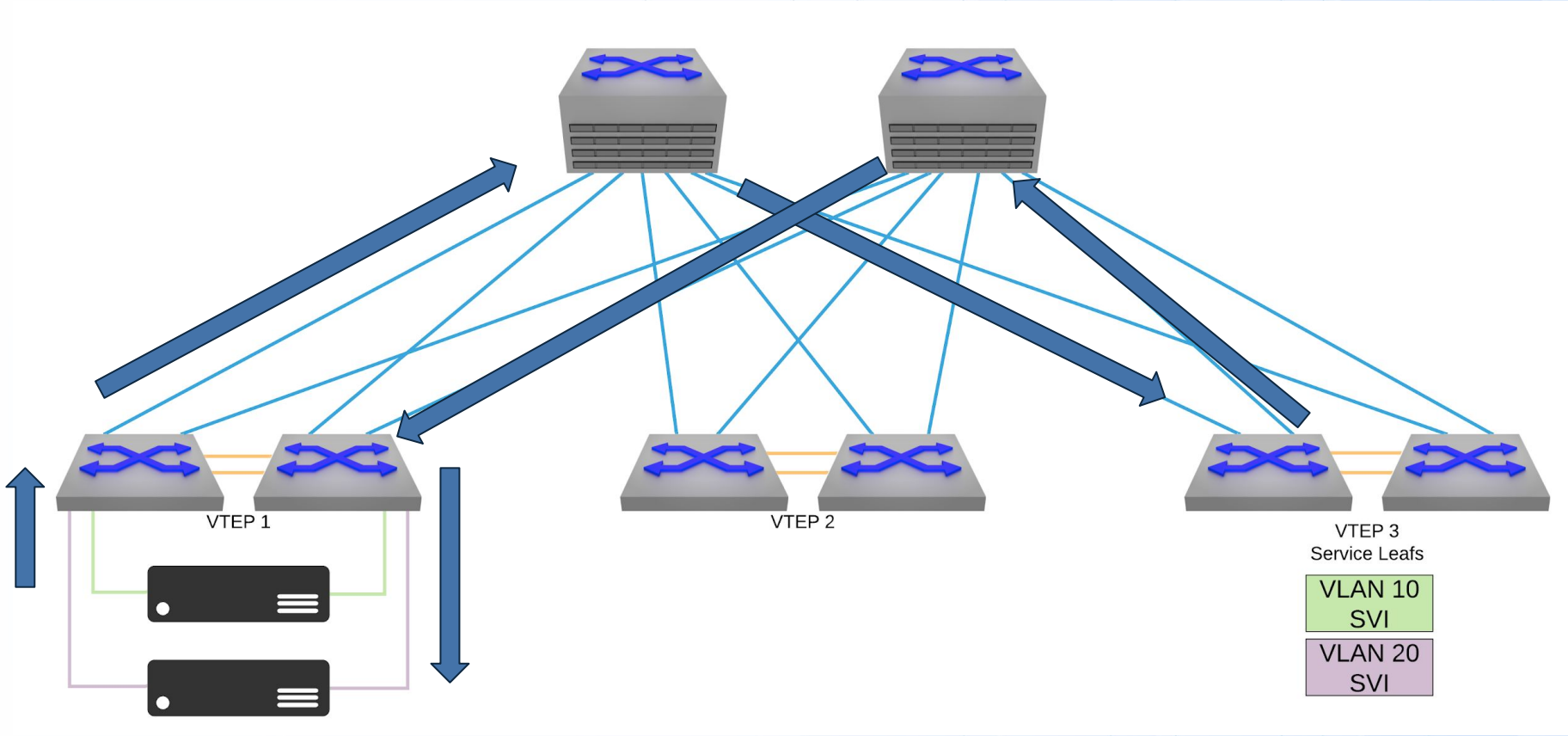
 |

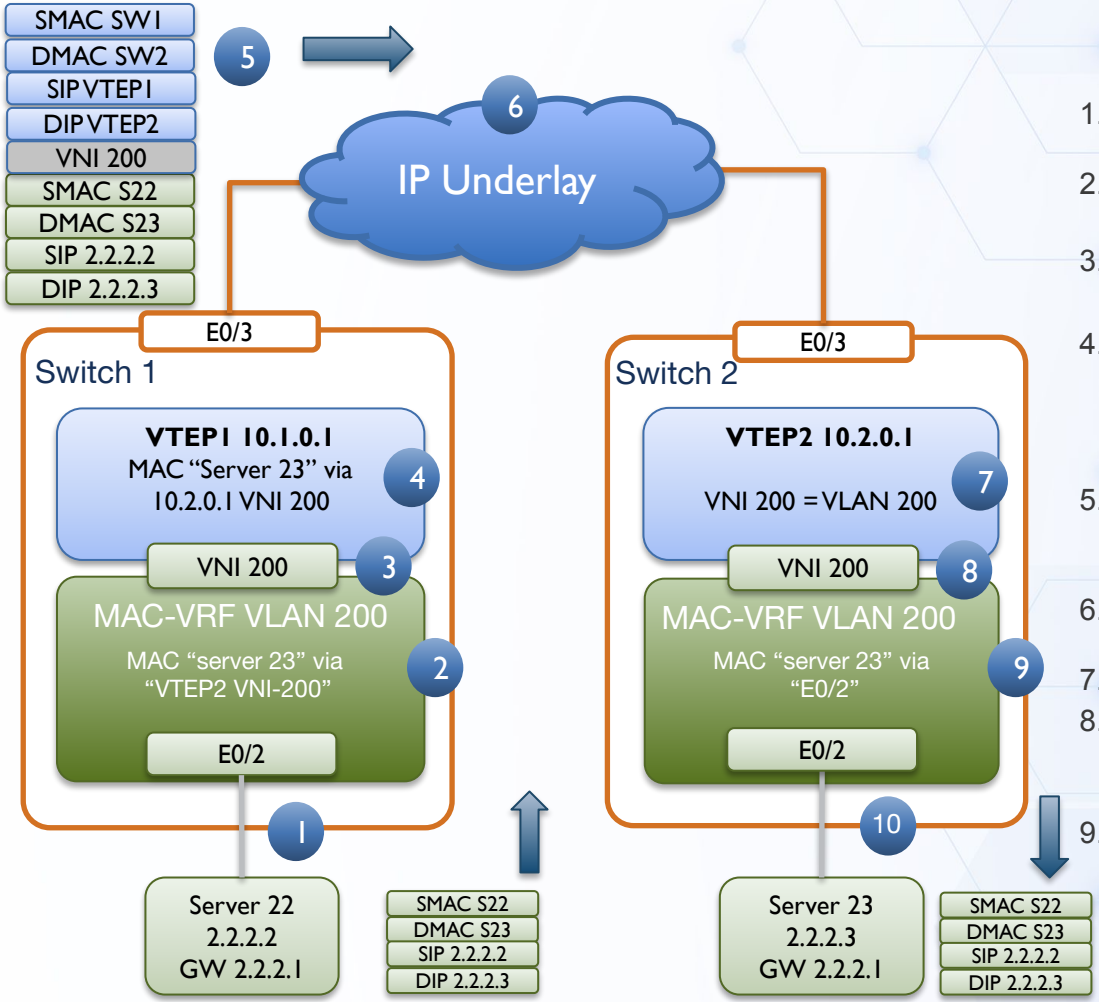
Traffic Flow - Different Leafs Different VLAN



 |

Traffic Flow - Same Leaf different VLAN





Server22 sends a packet to Server23

1. The L2 frame enters SW1 on port E0/2 VLAN 200 with a DMAC of "Server 23".
2. A L2 lookup is performed in the MAC-VRF VLAN 200
    1. The next hop is via VTEP2 VNI 200
3. The frame is switched into the VTEP via VNI 200.

    Think of VNI 200 as a virtual port
4. The VTEP adds the VXLAN header and outer UDP/IP header

    SIP = 10.1.0.1 (VTEP1)

    DIP = 10.2.0.1 (VTEP2)
5. The IP packet is routed over the IP underlay to SW2 VTEP2
6. VTEP2 remove the IP and VXLAN headers and reads the VNI field
7. The VTEP forwards the original L2 frame to VNI 200.
8. The packet is bridged according to the MAC table of the MAC-VRF VLAN 200

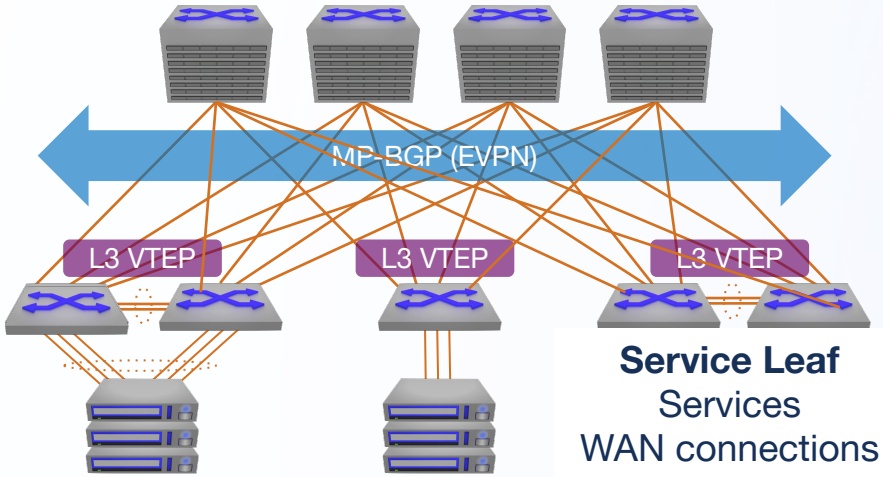
    In this case the packet is forwarded out E0-/2
9. The packet reaches server 23.

 |

IRB: Centralized Packet Capture

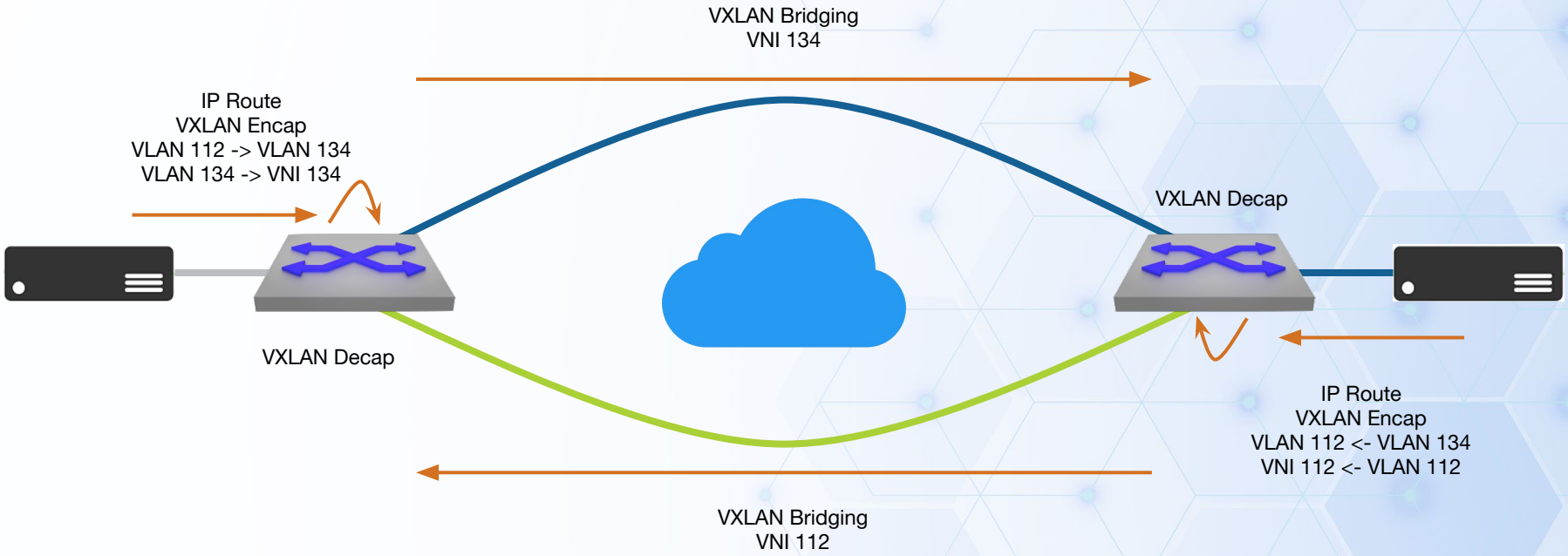
Prove it!

EVPN Asymmetric Routing



- Routing occurs on Leafs
- Anycast Gateway
- Every VLAN needs to be on every leaf
- Route Types Used
  - Type 2
    - MAC
    - MAC and IP
  - Type 3

Routing with Asymmetric IRB

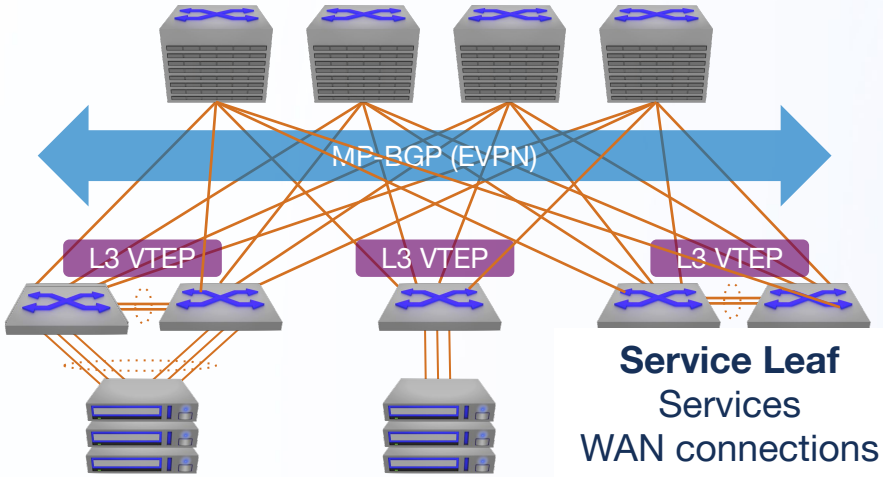


Original packet destination MAC address is host MAC address

IRB: Asymmetric Packet Capture

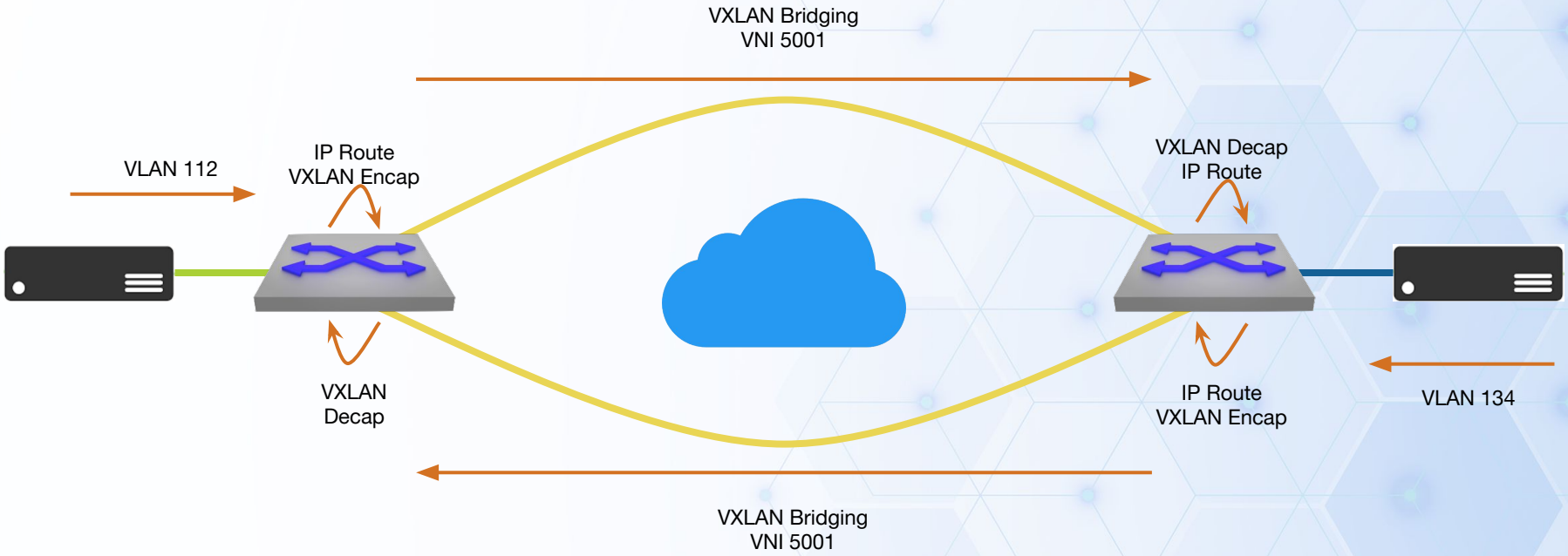
Prove it!

EVPN Symmetric Routing



- Routing occurs on Leafs
- Anycast Gateway
- Route Types Used
  - Type 2
    - MAC
    - MAC and IP
  - Type 3
  - Type 5

Routing with Symmetric IRB



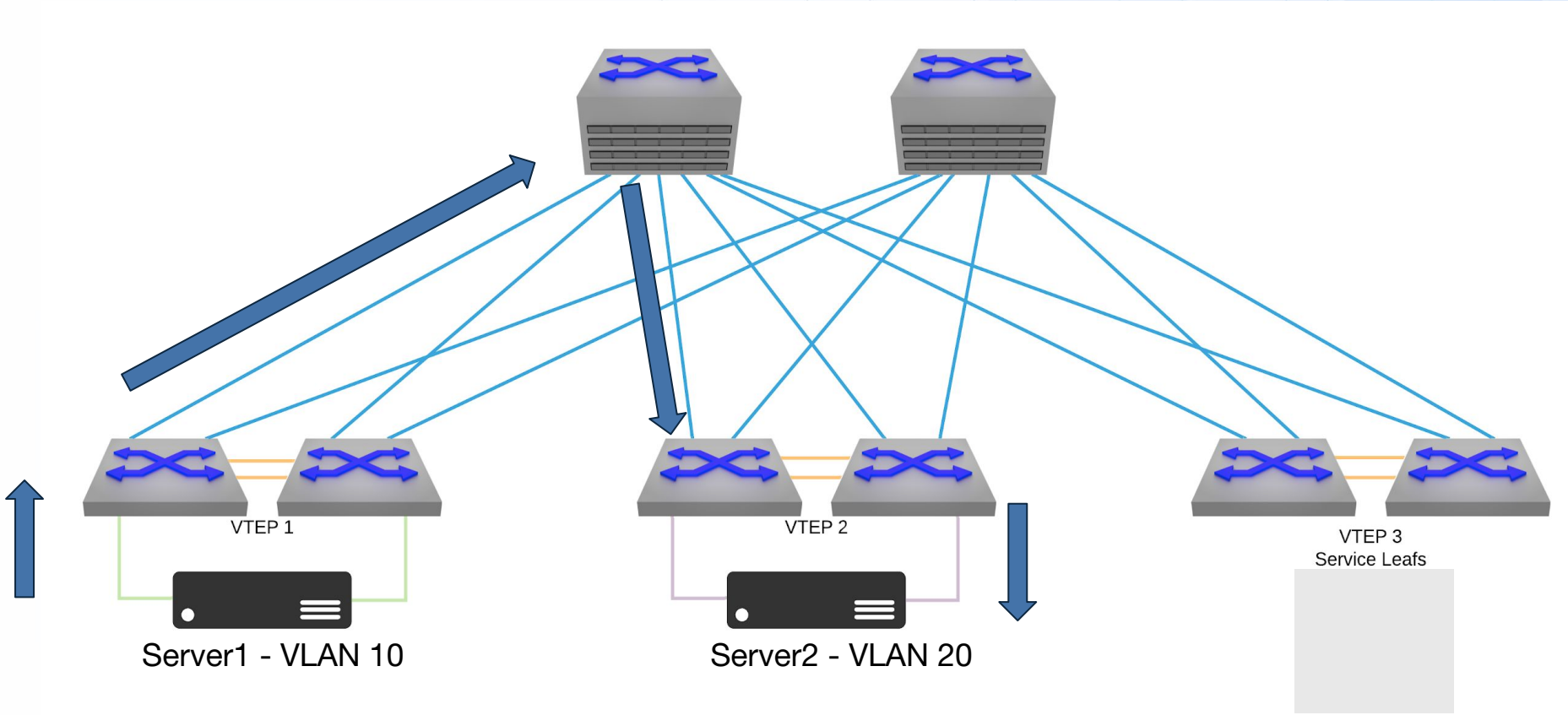
Original packet destination MAC address is router MAC address

IRB: Symmetric Packet Capture

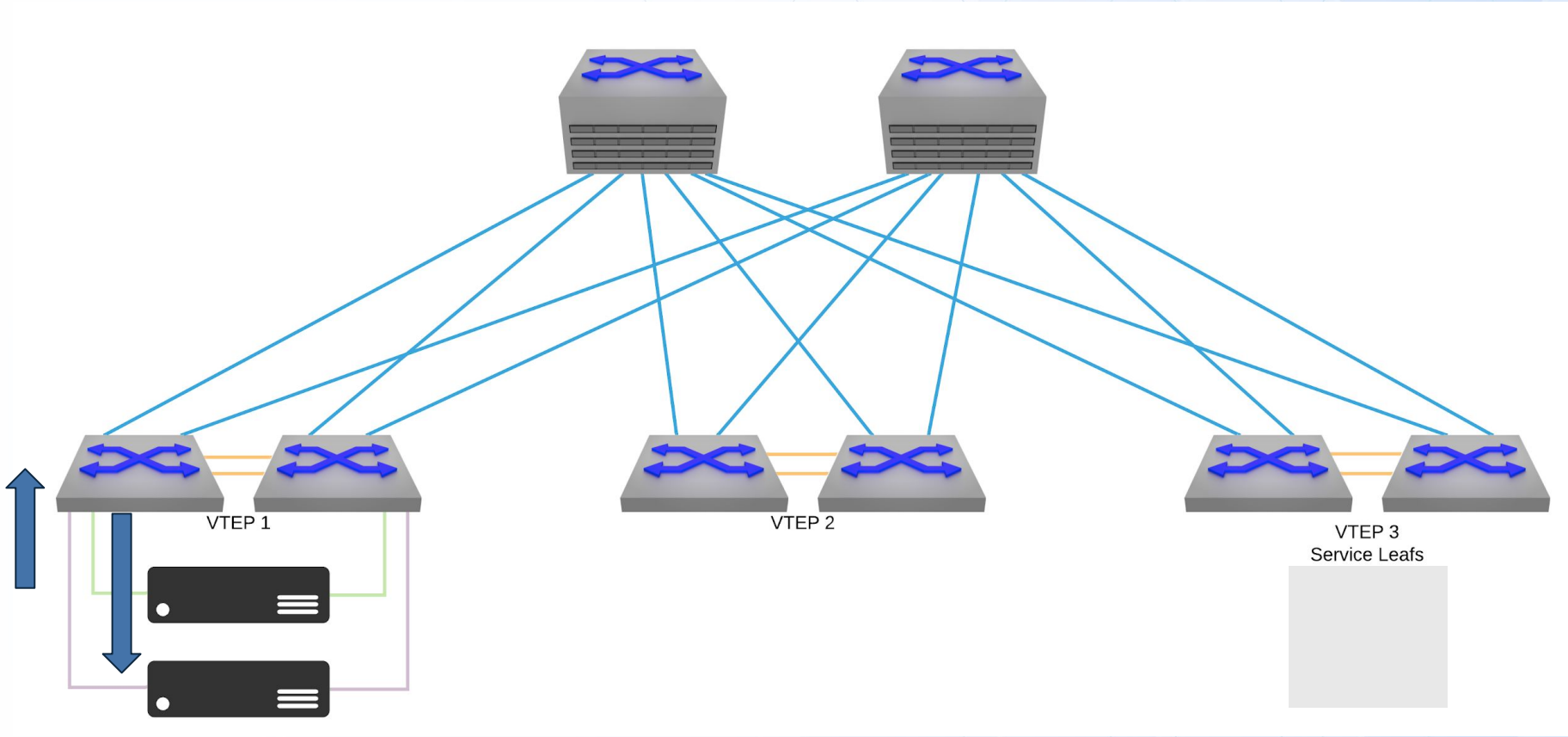
Prove it!

Traffic Flow IRB - Different Leafs Different VLAN

JS



Traffic Flow IRB - Same Leaf different VLAN



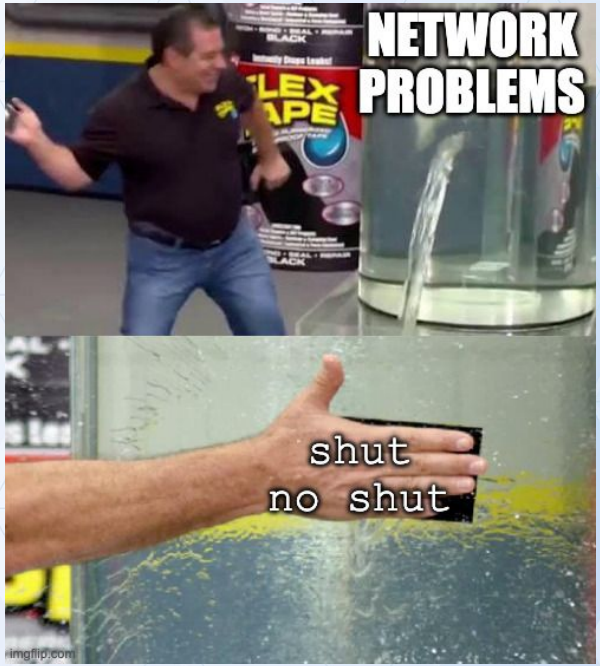
L3LS-V Zoom: L3 Routing

JS

- Common attributes:
  -

Troubleshooting EVPN:

- How do I use my existing troubleshooting tools & skills, within an EVPN network?
- MAC Moves/Blacklist
- Source-NAT
  - Traceroute & Ping
- MAC VRF's not created correctly or at all
- Incorrect L3 VNI
- Incorrect route-targets
- Extended Communities
- Router MAC Misconfigured



 |

VXLAN: Troubleshooting Commands

Troubleshooting: EVPN TS Commands:

1. show bgp evpn summary
   – Displays EVPN BGP neighbor status, session state, and basic statistics (similar to show ip bgp summary for the underlay).
2. show bgp evpn route-type mac-ip
   – Shows EVPN Type-2 routes (MAC + optional IP bindings) in the control plane.
3. show bgp evpn route-type mac-ip <MAC> detail
   – Gives detailed info about a specific MAC/IP EVPN route, including next-hops and communities.
4. show bgp evpn route-type ip-prefix ipv4
   – Lists EVPN Type-5 routes (IP prefix routes) for IPv4.
5. show bgp evpn route-type ip-prefix <SUBNET> detail
   – Provides detailed attributes for a specific IP prefix EVPN route.
6. show bgp neighbors x.x.x.x evpn received-routes detail
   – Displays all EVPN routes learned from a specific BGP neighbor, with full attributes.

Troubleshooting: EVPN TS Commands:

7. show bgp neighbors x.x.x.x evpn advertised-routes detail
   – Shows all EVPN routes this router advertised to a specific BGP neighbor, with attributes.
8. show vxlan config-sanity
   – Runs built-in checks for VXLAN/EVPN configuration consistency and highlights misconfigs.
9. show vxlan address-table
   – Displays the VXLAN MAC address table, mapping MACs to VNIs and VTEPs.
10. show vxlan vtep
    – Lists remote VXLAN Tunnel Endpoints (VTEPs) the switch has discovered via EVPN.
11. show vxlan vni
    – Shows configured VNIs and their associated VLANs/VRFs.
12. show arp remote
    – Displays ARP entries learned remotely via EVPN (not local ARP).

Troubleshooting: Existing Tools & Skills

Troubleshooting still boils down to classic networking skills and knowledge.

- IP Connectivity Checks (Underlay)
  - Can I ping my neighbor?
  - Can I ping the nexthop?
- Routing Troubleshooting (Underlay + Overlay)
  - Is peering up?
- L2 Troubleshooting (MAC Learning)
  - Where did I learn this MAC / Did I learn this MAC
  - Is the MAC moving

Troubleshooting: Source-NAT

- What is this?
- How does this apply to troubleshooting?
- Commands
  - `ip address virtual source-nat address <IP>`
  - `ip address virtual source-nat vrf <VRF> address <IP>`

More Discussion

- VTEP ICMP Enhancements for Improved Visibility
  - https://www.arista.com/en/support/toi/eos-4-33-2f/21106-vtep-icmp-enhancements-for-improved-visibility
-

Troubleshooting: MAC Moves/Naughty List

- Symptoms
  - MAC Address not in the remote MAC Table
  - Remote MAC table pointing at wrong VTEP
  - Seeing increased traffic on MLAG Peer Link
  - Seeing traffic for host on a different VTEP

Troubleshooting: MAC Moves/Naughty List

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present and correct
- `show bgp evpn host-flap` → MAC addresses on the naughty list

Troubleshooting: MAC Moves/Naughty List

Let’s the MACs move!

Troubleshooting:

Let the PINGs begin!!

Troubleshooting: Misconfigured Route-Targets

AO

Current Config:

```
s1-leaf1#
router bgp 65101
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   ...
```

```
s1-leaf2#
router bgp 65101
    vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   ...
```

What happens if we remove the VLAN route-target?

```
s1-leaf1#
router bgp 65101
   vlan 112
      no route-target both 112:112
   ...
```

```
s1-leaf2#
router bgp 65101
   vlan 112
      no route-target both 112:112
   ...
```


Troubleshooting: Misconfigured Route-Targets

AO

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show bgp evpn route-type mac-ip detail` →
- `show bgp evpn route-type ip-prefix x.x.x.x/x detail` →
- Show bgp evpn sanity detail

Troubleshooting: Misconfigured Route-Targets

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?
- Any differences between leaf pairs?

Checks

- What do the MAC / VXLAN address tables show?
- What do the MAC-IP (TYPE 2) routes show?
- What do the IMET (TYPE 3) routes show?

Troubleshooting: Misconfigured Route-Targets

AO

Let’s fix the issue:

```
s1-leaf1#
router bgp 65101
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   ...
```

```
s1-leaf2#
router bgp 65101
    vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
   ...
```

Validate things are working again

Troubleshooting: Communities


```
neighbor <PEER-GROUP> send-community standard
VS
neighbor <PEER-GROUP> send-community extended
VS
neighbor <PEER-GROUP> send-community standard extended
VS
neighbor <PEER-GROUP> send-community
```


Troubleshooting: Misconfigured Communities

Current Config:

s1-leaf1#

```
router bgp 65101
   ...
   neighbor MLAG send-community standard extended
   neighbor SPINE peer group
   neighbor SPINE remote-as 65100
   neighbor SPINE send-community standard extended
   neighbor SPINE-EVPN peer group
   ...
   neighbor SPINE-EVPN send-community standard extended
```

s1-leaf2#

```
router bgp 65101
   ...
   neighbor MLAG send-community standard extended
   neighbor SPINE peer group
   neighbor SPINE remote-as 65100
   neighbor SPINE send-community standard extended
   neighbor SPINE-EVPN peer group
   ...
   neighbor SPINE-EVPN send-community standard extended
```

What happens if we remove the extended community from the overlay?

s1-leaf1#

```
router bgp 65101
   ...
   neighbor SPINE-EVPN send-community standard
```

s1-leaf1#

```
router bgp 65101
   ...
   neighbor SPINE-EVPN send-community standard
```


Troubleshooting: Misconfigured Communities

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show vxlan address-table` → src and/or dst MAC addresses present learned via VXLAN
- `show bgp neighbors x.x.x.x evpn advertised-routes detail`→
- `show bgp neighbors x.x.x.x evpn received-routes detail`→
- Show bgp evpn sanity detail

Troubleshooting: Misconfigured Communities

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?
- Any differences between leaf pairs?

Checks

- What do the MAC / VXLAN address tables show?
- What do the MAC-IP (TYPE 2) routes show?
- What do the IMET (TYPE 3) routes show?

Troubleshooting: Misconfigured Communities

Let’s fix the issue:

```
s1-leaf1#
router bgp 65101
   ...
   neighbor SPINE-EVPN send-community standard extended
```

```
s1-leaf2#
router bgp 65101
   ...
   neighbor SPINE-EVPN send-community standard extended
```

Validate things are working again

Troubleshooting: VxLAN Source Interface

AO

Current Config:

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

What happens if we remove the extended community from the overlay?

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback0
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback0
```


Troubleshooting: VxLAN Source Interface

AO

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show interface vxlan1` → status, source interface, encapsulation MAC, flood list, VNI mapping
- `show mac address-table` → src and/or dst MAC addresses present
- `show vxlan flood vtep` →

Troubleshooting: VxLAN Source Interface

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?

Checks

- What do the MAC / VXLAN address tables show?
- What do the MAC-IP (TYPE 2) routes show?
- What do the IMET (TYPE 3) routes show?

Troubleshooting: VxLAN Source Interface

AO

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

Validate things are working again

ARISTA

Troubleshooting: Router MAC

Current Config:

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address mlag-system-id
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address mlag-system-id
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

What happens if we remove the extended community from the overlay?

s1-leaf1#

```
interface Vxlan1
   vxlan source-interface Loopback1
   no vxlan virtual-router encapsulation mac-address mlag-system-id
```

s1-leaf2#

```
interface Vxlan1
   vxlan source-interface Loopback1
   no vxlan virtual-router encapsulation mac-address mlag-system-id
```


Troubleshooting: Router MAC

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- `show ip route vrf TENANT` →
- `show bgp evpn route-type mac-ip detail` →
- `show bgp evpn route-type ip-prefix x.x.x.x/x detail` →

Troubleshooting: Router MAC

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?

Checks

- What do the MAC / VXLAN address tables show?
- What do the MAC-IP (TYPE 2) routes show?
- What do the IP-PREFIX (TYPE 5) routes show?
- What does the routing table for VRF TENANT show?

Troubleshooting: Router MAC

Let’s fix the issue:

```
s1-leaf1#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

```
s1-leaf2#
interface Vxlan1
   vxlan source-interface Loopback1
   vxlan virtual-router encapsulation mac-address…
   vxlan udp-port 4789
   vxlan vlan 112 vni 112
   vxlan vlan 134 vni 134
   vxlan vrf TENANT vni 5001
```

Validate things are working again

Troubleshooting: MAC VRF

Current Config:

s1-leaf1#

```
router bgp 65101
   …
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
```

s1-leaf2#

```
router bgp 65101
   …
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
```

What happens if we remove a MAC VRF?

s1-leaf1#

```
router bgp 65101
   …
   no vlan 112
```

s1-leaf2#

```
router bgp 65101
   …
   no vlan 112
```


Troubleshooting: MAC VRF

Troubleshooting Tools:

- `ping host to host` → packet loss between end-hosts?
- `show mac address-table` → src and/or dst MAC addresses present
- Show int vxlan1
- Sho bgp nei received routes
- Sho bgp nei adv routes

Troubleshooting: MAC VRF

What just happened??

- Did anything break?
- Why or why not?
- What can we look at?

Checks

- What do the MAC / VXLAN address tables show?
- What do the MAC-IP (TYPE 2) routes show?
- What do the IMET (TYPE 3) routes show?

Troubleshooting: MAC VRF

Let’s fix the issue:

```
s1-leaf1#
router bgp 65101
   …
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
```

```
s1-leaf2#
router bgp 65101
   …
   vlan 112
      rd auto
      route-target both 112:112
      redistribute learned
```

Validate things are working again

Troubleshooting: L3 VNI

- When is this important and when is it not?
- What will the switch do with it missing?
- Can they be different between switches?
- What happens?

Thank you!

2025 Arista EVPN Workshop

