

VXLAN

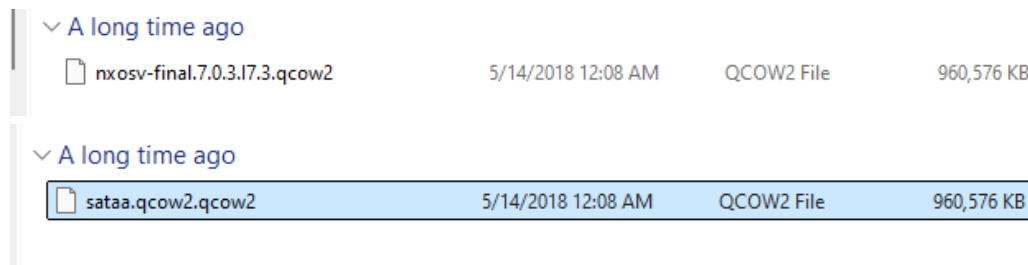
## TABLE OF CONTENTS

การติดตั้ง Nexus 9000v บน EVE NG	3
การทดลอง Lab	8
คำถามอภิปราย	15
ผลสรุป	19

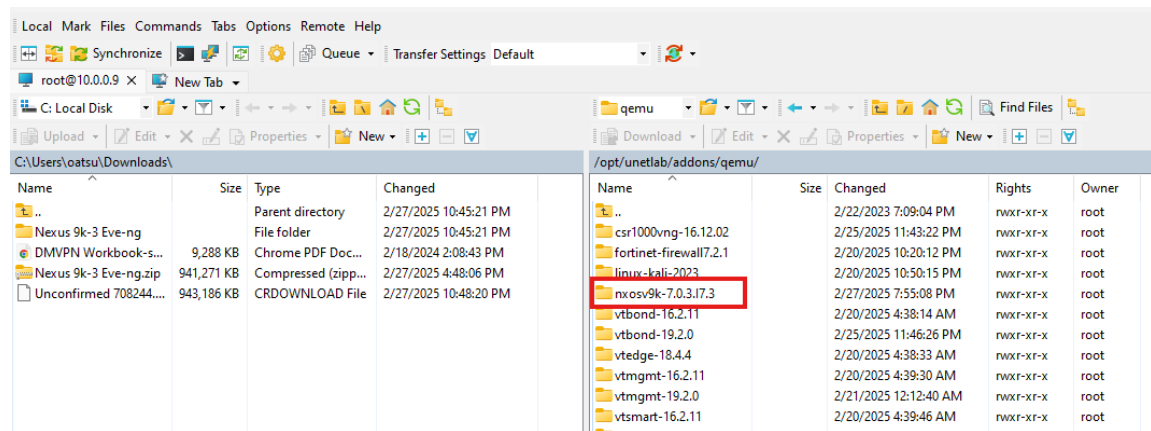
## 1. ดาวน์โหลด Image

[https://mega.nz/file/8d11kaAI#\\_Ovfp\\_1yI4cxCPfwyO\\_J41bBBpedRPZL3l8KQmxSExA](https://mega.nz/file/8d11kaAI#_Ovfp_1yI4cxCPfwyO_J41bBBpedRPZL3l8KQmxSExA)

## 2. ทำการแตกไฟล์ จากนั้นเปลี่ยนชื่อ Image เป็น sataa.qcow2



## 3. อัปโหลดเข้าไปยัง EVE-NG โดยตั้งชื่อไฟล์เดอรัว่า nxosv9k-7.0.3.i7.3

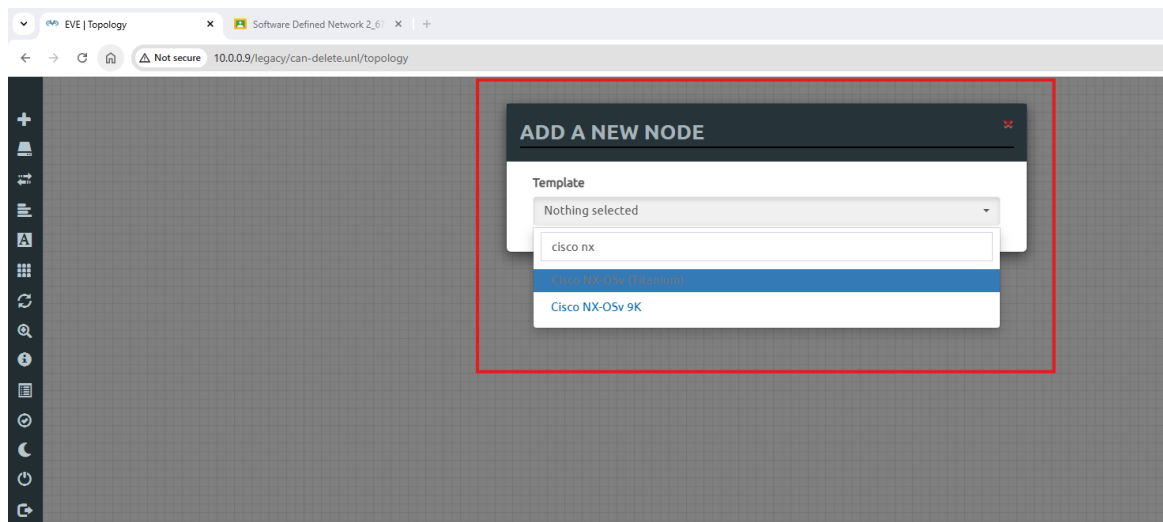


## 4. ใช้คำสั่งนี้ที่ eve-server

```
/opt/unetlab/wrappers/unl_wrapper -a fixpermissions
```

```
Last login: Thu Feb 27 12:55:17 2025 from 10.0.0.2
root@eve-ng:~#
root@eve-ng:~#
root@eve-ng:~# /opt/unetlab/wrappers/unl_wrapper -a fixpermissions
```

5. ทดลองเพิ่ม Node ที่ eve โดยค้นหา Node ที่ชื่อว่า Cisco NX-OSv 9K



### ADD A NEW NODE

**Template**  
Cisco NX-OSv 9K

**Number of nodes to add**  
1

**Image**  
nxosv9k-7.0.3.17.3

**Name/prefix**  
NXOS

**Icon**  
Nexus7K.png

**UUID**

**CPU Limit** ☐

**CPU**  
2

**RAM (MB)**  
8192

**Ethernets**  
8

**QEMU Version**  
tpl(4.1.0)

**QEMU Arch**  
tpl(x86\_64)

**QEMU Nic**  
tpl(e1000)

**QEMU custom options**  
-machine type=pc,accel=kvm -serial mon:stdio -nographic -enable-kvm

**Startup configuration**  
None

**Delay (s)**  
0

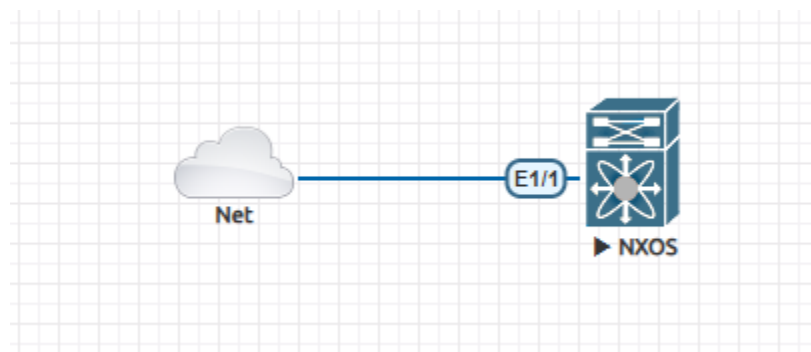
**Console**  
telnet

**Left**  
601

**Top**  
219

**Save** **Cancel**

6. เพิ่ม Network (Management) แล้วเชื่อมเข้ากับ Nexus จากนั้นทำการ Start Node



```

NXOS
Linking n9k flash devices
INIT: version 2.88 booting
Installing ata_piix module ... done.
Unsquashing rootfs ...
Installing isan procfs ... done.
Installing SSE module with card index 21099 ... done.
Creating SSE device node 246 ... done.
Loading I2C driver ... done.
Installing CTRL driver for card_type 29 on a VM without NEED
19.56: Interrupt throttling disabled. No ctrl irq detected.
Loading IGB driver ... done.
Not Micron SSD...

Checking all filesystems.Warning: switch is starting up with
on

Installing SPROM driver ... IS_N9K done.
Installing pfmsvcs module with SPROM card index 21099 ... don
Installing nvram module ... done.
Installing if_index module with port mode 6 ... done.
Installing fcFwd
Installing RNI lcnd ... done
Installing LC netdev ... done

```

7. ในกรณีที่ Node เข้าหน้า loader> นั่นคืออุปกรณ์หา Image ที่จะใช้ boot ไม่เจอ ให้ใช้คำสั่งนี้ที่ Node

```
loader> boot bootflash:nxos.7.0.3.17.3.bin
```

```

NXOS
loader > dir bootflash: 1
Error 30: Invalid argument

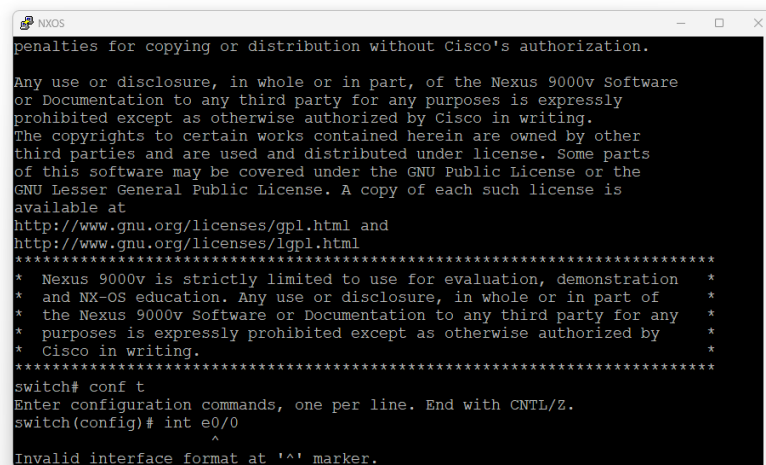
loader > dir bootflash:
2 Setting listing for bootflash:
Number of devices detected by BIOS is 1
Number of devices detected by BIOS is 1
Number of devices detected by BIOS is 1
Going to print files for device bootflash: 1
.rpmsstore
nxos.7.0.3.17.3.bin
Number of devices detected by BIOS is 1
Number of devices detected by BIOS is 1
Number of devices detected by BIOS is 1
Clearing listing for bootflash:. Ret 0

loader > boot bootflash:nxos.7.0.3.17.3.bin
Booting bootflash:nxos.7.0.3.17.3.bin
Trying diskboot
dev_str: bootflash: partition_str: 3 filename_str /nxos.7.0.3.17.3.bin
command = root (hd0,3)
Filesystem type is ext2fs, partition type 0x83
Formed cmdline console=ttyS0,115200n8nn loader ver="5.01.0" quiet debug

```

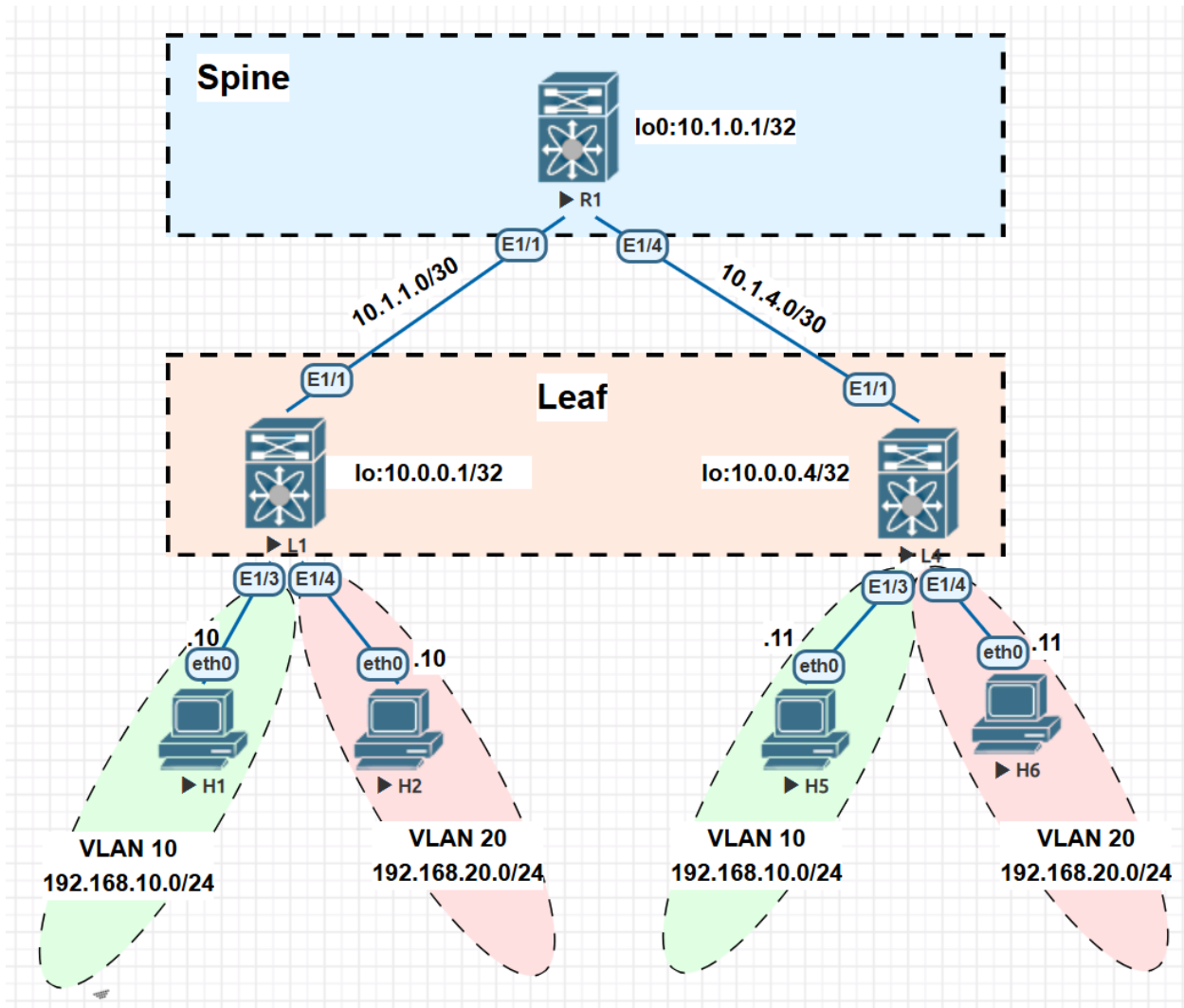
8. หลังจากอุปกรณ์พร้อมแล้ว ตัวอุปกรณ์จะถามคำถามต่อไปนี้ ให้ตอบดังนี้

- a. Abort Power On Auto Provisioning [yes - continue with normal setup, skip - bypass password and basic configuration, no - continue with Power On Auto Provisioning]  
(yes/skip/no)[no]: yes
- b. Do you want to enforce secure password standard (yes/no) no
- c. Enter passwords for admin:  
  
admin  
  
admin
- d. Would you like to enter the basic configuration dialog (yes/no): no



```
penalties for copying or distribution without Cisco's authorization.
Any use or disclosure, in whole or in part, of the Nexus 9000v Software
or Documentation to any third party for any purposes is expressly
prohibited except as otherwise authorized by Cisco in writing.
The copyrights to certain works contained herein are owned by other
third parties and are used and distributed under license. Some parts
of this software may be covered under the GNU Public License or the
GNU Lesser General Public License. A copy of each such license is
available at
http://www.gnu.org/licenses/gpl.html and
http://www.gnu.org/licenses/lgpl.html
*****
* Nexus 9000v is strictly limited to use for evaluation, demonstration *
* and NX-OS education. Any use or disclosure, in whole or in part of *
* the Nexus 9000v Software or Documentation to any third party for any *
* purposes is expressly prohibited except as otherwise authorized by *
* Cisco in writing. *
*****
switch# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# int e0/0
      ^
Invalid interface format at '^' marker.
```

# BGP-EVPN VXLAN Lab



## Topology

เราจะสร้างโทโพโลยีตามแบบที่กำหนดไว้ แนะนำให้ตั้งชื่ออินเทอร์เฟซและเชื่อมต่ออุปกรณ์เหมือนในโทโพโลยีเลย จะได้ตรงกันเวลาทำ LAB ตั้งค่า IP ให้เหมือนกับที่กำหนดไว้ ห้ามเปลี่ยน โทโพโลยีนี้แบ่งเป็นเลเยอร์ได้แก่ Spine, Leaf และ End Host ตามที่เห็นในแผนภาพ



Spine และ Leaf ทุกโหนดเป็น NXOSv L3 Switch ดังนั้นต้องติดตั้ง NXOSv ทั้งหมด 3 โหนด ได้แก่ R1, L1 และ L4

End-Hosts สามารถใช้โหนดอะไรก็ได้ที่รองรับฟังก์ชันเครือข่ายพื้นฐาน เช่น ping และ ARP แนะนำให้ใช้ VPCS ซึ่งมีมาให้ใน EVE-NG โดยไม่ต้องติดตั้งเพิ่ม

- H1 และ H2 เชื่อมต่ออยู่หลัง Leaf L1
- H5 และ H6 เชื่อมต่ออยู่หลัง Leaf L4

ลิงก์ทั้งหมดระหว่าง Spine R1 และ Leaf L1 กับ L4 จะเป็น Layer 3 เราต้องตั้งค่าทั้ง Unicast และ Multicast Configuration โดยเริ่มจาก Unicast ก่อน

## Unicast Configuration (ต้องทำบน Spine และ Leaf Nodes)

เป้าหมายของการตั้งค่า Unicast คือการทำให้ทุกโหนดในเครือข่ายสามารถเข้าถึงกันได้

- กำหนดค่า IP Address บนทุกอินเทอร์เฟซที่เชื่อมต่อระหว่าง Spine และ Leaf
- ตั้งค่า Loopback Address 0 บนทุกโหนดโดยใช้ /32
- ใช้ Routing Protocol เช่น OSPF หรือ ISIS ระหว่าง Spine และ Leaf เพื่อให้ทุกโหนดสามารถเข้าถึงกันได้

Spine R1	Leaf L1	Leaf L4
<pre> conf hostname spine1 feature ospf router ospf SI-Underlay router-id 10.1.0.1 end  conf interface loopback 0 no sh ip address 10.1.0.1/32 ip router ospf SI-Underlay area 0 </pre>	<pre> conf hostname leaf1 feature ospf router ospf SI-Underlay router-id 10.0.0.1 end  conf interface loopback 0 no sh ip address 10.0.0.1/32 ip router ospf SI-Underlay area 0 </pre>	<pre> conf hostname leaf4 #boot nxos nxos.10.1.1.bin feature ospf router ospf SI-Underlay router-id 10.0.0.4 end  conf interface loopback 0 no sh ip address 10.0.0.4/32 </pre>

<pre> ip ospf network point-to-point end  conf interface ethernet 1/1 description to-leaf1-eth1/1 no sh no sw ip address 10.1.1.2/30 ip router ospf SI-Underlay area 0 ip ospf network point-to-point mtu 9000  interface ethernet 1/4 description to-leaf4-eth1/1 no sw ip address 10.1.4.2/30 ip router ospf SI-Underlay area 0 ip ospf network point-to-point mtu 9000 no sh end </pre>	<pre> ip ospf network point-to-point end conf interface ethernet 1/1 description to-spine1-eth1/0 no sh no sw ip address 10.1.1.1/30 ip router ospf SI-Underlay area 0 ip ospf network point-to-point mtu 9000 </pre>	<pre> ip router ospf SI-Underlay area 0 ip ospf network point-to-point end  conf interface ethernet 1/1 description to-spine1-eth1/3 no sh no sw ip address 10.1.4.1/30 ip router ospf SI-Underlay area 0 ip ospf network point-to-point mtu 9000 </pre>
--	---	--

**Verification:**

ทดสอบ Ping ไปยัง Loopback Address ของแต่ละโหนด (ระหว่าง R1, L1 และ L4) จากโหนดใดก็ได้ หากสามารถ Ping ได้ครบทุกโหนด แสดงว่าการตั้งค่า Unicast Routing Protocol เสร็จสมบูรณ์

```

leaf1#
leaf1# ping 10.1.0.1
PING 10.1.0.1 (10.1.0.1): 56 data bytes
64 bytes from 10.1.0.1: icmp_seq=0 ttl=254 time=54.032 ms
64 bytes from 10.1.0.1: icmp_seq=1 ttl=254 time=370.957 ms
64 bytes from 10.1.0.1: icmp_seq=2 ttl=254 time=13.593 ms
64 bytes from 10.1.0.1: icmp_seq=3 ttl=254 time=61.997 ms
64 bytes from 10.1.0.1: icmp_seq=4 ttl=254 time=19.457 ms

--- 10.1.0.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 13.593/104.007/370.957 ms
leaf1#

```

## Multicast Configuration

เป้าหมายของการตั้งค่า Multicast คือการกระจายทราฟฟิกประเภท BUM (Broadcast, Unknown Unicast, Multicast)

- เปิดใช้งาน PIM Sparse Mode บนทุกอินเทอร์เฟซที่เชื่อมต่อระหว่าง Spine และ Leaf
- กำหนดให้ Spine เป็น Rendezvous Point (RP)
- กำหนดให้ Leafs (L1 และ L4) เป็น Last Hop Router (LHR) สำหรับ Multicast Group 239.1.1.1 และ 239.2.2.2

Spine R1	Leaf L1	Leaf L4
<pre> conf feature pim ip pim rp-address 10.100.100.100 ip pim anycast-rp 10.100.100.100 10.1.0.1 ip pim anycast-rp 10.100.100.100 10.1.0.2 interface loopback 0 ip pim sparse-mode interface loopback 1 ip pim sparse-mode description anycast-rp ip address 10.100.100.100/32 </pre>	<pre> conf feature pim ip pim rp-address 10.100.100.100 interface loopback 0 ip pim sparse-mode interface ethernet 1/1 ip pim sparse-mode </pre>	<pre> conf feature pim ip pim rp-address 10.100.100.100 interface loopback 0 ip pim sparse-mode interface ethernet 1/1 ip pim sparse-mode </pre>

<pre> no sh ip router ospf SI-Underlay area 0 interface ethernet 1/1 ip pim sparse-mode interface ethernet 1/4 ip pim sparse-mode end </pre>		
--	--	--

## VxLAN Configuration ( Overlay Configuration )

ตอนนี้เราต้องตั้งค่า VxLAN เบื้องต้น ซึ่งต้องทำเฉพาะบน Leaf Switches (L1 และ L4) เท่านั้น  
หลังจากตั้งค่า VxLAN เสร็จแล้ว End Hosts ที่อยู่ใน VLAN เดียวกันจะต้องสามารถ Ping หากันได้

### Leaf L1

### Leaf L4

<pre> conf feature interface-vlan feature nv overlay feature vn-segment-vlan-based interface nve 1 no sh source loopback 0 member vni 5010 mcast-group 239.1.1.1 member vni 5020 mcast-group 239.2.2.2 vlan 10 vn-segment 5010 vlan 20 vn-segment 5020 interface ethernet 1/3 sw mode access sw access vlan 10 interface ethernet 1/4 sw mode access sw access vlan 20 </pre>	<pre> conf feature interface-vlan feature nv overlay feature vn-segment-vlan-based interface nve 1 no sh source loopback 0 member vni 5010 mcast-group 239.1.1.1 member vni 5020 mcast-group 239.2.2.2 vlan 10 vn-segment 5010 vlan 20 vn-segment 5020 interface ethernet 1/3 sw mode access sw access vlan 10 interface ethernet 1/4 sw mode access </pre>
---	---

end	sw access vlan 20 end
-----	--------------------------

### Verification:

ทดสอบ Ping จากแต่ละ End Host ตามนี้:

- จาก H1, Ping ไปที่ 192.168.10.11
- จาก H2, Ping ไปที่ 192.168.20.11
- จาก H5, Ping ไปที่ 192.168.20.10
- จาก H6, Ping ไปที่ 192.168.10.10

หากทุก Ping สำเร็จ แสดงว่าการตั้งค่า VxLAN ทำงานได้ถูกต้อง โดยตอนนี้ VxLAN รองรับเฉพาะการสื่อสารภายใน VLAN เดียวกัน (Intra-VLAN Routing) ซึ่งหมายความว่า VLAN ถูกขยายออกไป เช่น VLAN 10 มีอยู่ทั้งหลัง Leaf L1 และ L4 และ VLAN 20 ก็เช่นกัน

นอกจากนี้ ทราฟฟิกประเภท BUM ถูกจัดการผ่าน Multicast แทนที่จะใช้ Ingress Replication ซึ่งมีประสิทธิภาพน้อยกว่าในการจัดการ BUM Traffic หากใน Data Center มี Leaf Switch 1,000 ตัว (ซึ่งเรียกว่า VTEP - Virtual Tunneling End Points) ก็สามารถมี VLAN 10 อยู่ 1,000 Segment โดยแต่ละ Segment อยู่หลัง Leaf หนึ่งตัว และทุก Host ใน VLAN เหล่านั้นสามารถสื่อสารกันได้ราวกับอยู่ใน VLAN เดียวกันผ่าน L2VNI

### Ping H1 -> H5

```

H1C
VPCS>
VPCS>
VPCS>
VPCS>
VPCS> show ip

NAME       : VPCS[1]
IP/MASK    : 192.168.10.10/24
GATEWAY    : 192.168.10.1
DNS        :
MAC        : 00:50:79:66:68:04
LPORT     : 20000
RHOST:PORT : 127.0.0.1:30000
MTU        : 1500

VPCS> ping 192.168.10.11

84 bytes from 192.168.10.11 icmp_seq=1 ttl=64 time=8.882 ms
84 bytes from 192.168.10.11 icmp_seq=2 ttl=64 time=8.491 ms
84 bytes from 192.168.10.11 icmp_seq=3 ttl=64 time=9.269 ms
84 bytes from 192.168.10.11 icmp_seq=4 ttl=64 time=15.659 ms
84 bytes from 192.168.10.11 icmp_seq=5 ttl=64 time=19.694 ms

VPCS>

```

Ping H2 -> H6

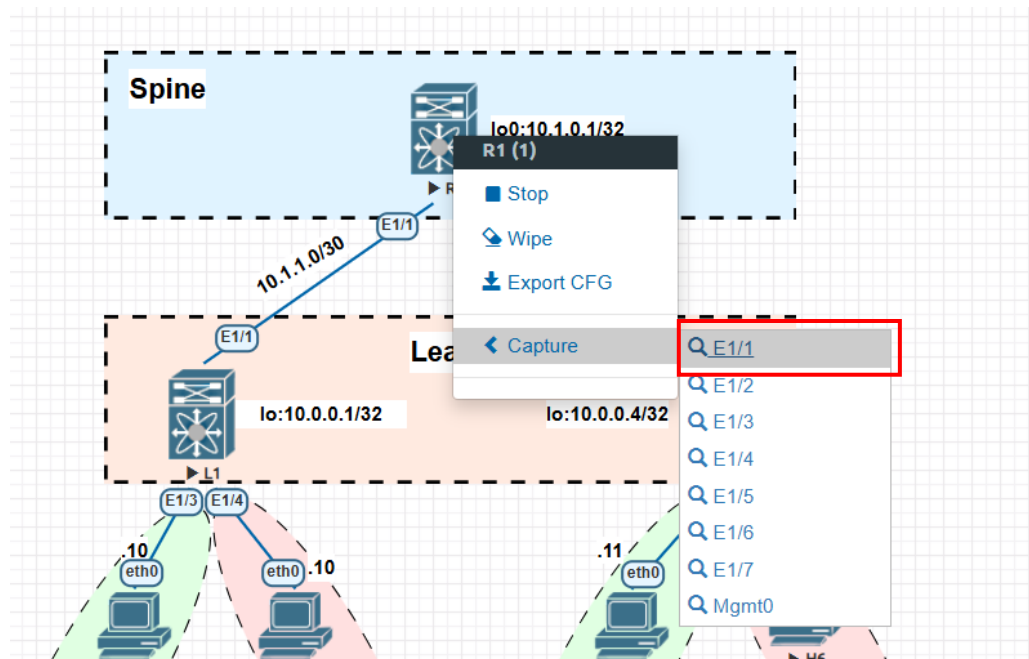
```
H2C
VPCS> show ip
NAME       : VPCS[1]
IP/MASK    : 192.168.20.10/24
GATEWAY    : 192.168.20.1
DNS        :
MAC        : 00:50:79:66:68:07
LPORT     : 20000
RHOST:PORT : 127.0.0.1:30000
MTU        : 1500

VPCS> ping 192.168.20.11

84 bytes from 192.168.20.11 icmp_seq=1 ttl=64 time=9.716 ms
84 bytes from 192.168.20.11 icmp_seq=2 ttl=64 time=10.627 ms
84 bytes from 192.168.20.11 icmp_seq=3 ttl=64 time=9.744 ms
84 bytes from 192.168.20.11 icmp_seq=4 ttl=64 time=9.720 ms
84 bytes from 192.168.20.11 icmp_seq=5 ttl=64 time=9.944 ms

VPCS>
```

Packet Capture – Spine1(E1/1)



Capturing from Standard input

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
15	14.629037	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0
16	14.634562	192.168.20.11	192.168.20.10	ICMP	148	Echo (ping) reply id=0
17	15.041603	50:00:00:02:00:07	CDP/VTP/DTP/PagP/UD...	CDP	238	Device ID: leaf1(9HTF3ZPN
18	15.354924	10.1.1.1	224.0.0.5	OSPF	82	Hello Packet
19	15.639778	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0
20	15.645124	192.168.20.11	192.168.20.10	ICMP	148	Echo (ping) reply id=0
21	16.650980	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0
22	16.656731	192.168.20.11	192.168.20.10	ICMP	148	Echo (ping) reply id=0
23	20.630830	10.1.1.2	224.0.0.5	OSPF	82	Hello Packet
24	24.194037	10.1.1.2	224.0.0.13	PIMv2	64	Hello
25	24.886356	10.1.1.1	224.0.0.5	OSPF	82	Hello Packet
26	25.098181	10.1.1.1	224.0.0.1	IGMPv2	60	Membership Query, general
27	29.248799	10.1.1.2	224.0.0.5	OSPF	82	Hello Packet
28	32.282197	10.1.1.1	224.0.0.13	PIMv2	64	Hello
29	32.446051	50:00:00:01:00:08	CDP/VTP/DTP/PagP/UD...	CDP	223	Device ID: spine1(9RIP2ZE
30	33.358409	10.1.1.1	224.0.0.5	OSPF	82	Hello Packet
31	37.624317	10.1.1.2	224.0.0.5	OSPF	82	Hello Packet

> Frame 12: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface -, id 0

> Ethernet II, Src: 50:00:00:01:00:07 (50:00:00:01:00:07), Dst: 50:00:00:02:00:07 (50:00:00:02:00:07)

> Internet Protocol Version 4, Src: 10.0.0.4, Dst: 10.0.0.1

> User Datagram Protocol, Src Port: 58441, Dst Port: 4789

> Virtual eXtensible Local Area Network

> Flags: 0x0800, VXLAN Network ID (VNI) Group Policy ID: 0

> VXLAN Network Identifier (VNI): 5020

> Reserved: 0

> Ethernet II, Src: Private\_66:68:06 (00:50:79:66:68:06), Dst: Private\_66:68:07 (00:50:79:66:68:07)

> Internet Protocol Version 4, Src: 192.168.20.11, Dst: 192.168.20.10

> Internet Control Message Protocol

Standard input: <live capture in progress> | Packets: 31 | Profile: Default

## 1. VXLAN คืออะไร และทำงานอย่างไรในเครือข่าย Layer 2 Overlay?

VXLAN (Virtual eXtensible LAN) เป็นเทคโนโลยีที่ช่วยขยายเครือข่าย Layer 2 ให้สามารถทำงานข้าม Layer 3 ได้โดยใช้ encapsulation ผ่าน UDP. VXLAN ใช้ VXLAN Network Identifier (VNI) ขนาด 24-bit เพื่อแทน VLAN ID ทำให้สามารถรองรับเครือข่ายเสมือน (Virtual Network) ได้มากถึง 16 ล้านเครือข่าย (เทียบกับ VLAN ที่มีได้แค่ 4096)

VXLAN อาศัยอุปกรณ์ที่เรียกว่า VTEP (VXLAN Tunnel End Point) ซึ่งทำหน้าที่ encapsulate และ decapsulate VXLAN packet. ในกรณีนี้ Leaf switch ทำหน้าที่เป็น VTEP โดย encapsulate traffic ระหว่าง VLAN 10 และ VLAN 20 ให้สามารถวิ่งข้าม Spine ไปยังอีก Leaf ได้

## 2. Packet ที่แสดงใน Wireshark มี VXLAN Network Identifier (VNI) เป็น 5020 หมายถึงอะไร?

VNI 5020 ใน Wireshark Capture ระบุว่า Packet นี้เป็น VXLAN frame ที่ encapsulate traffic ของ VLAN ที่อยู่ใน Virtual Network ID 5020. จาก Topology, VLAN 20 ใช้ subnet 192.168.20.0/24, และ Host 192.168.20.10 กับ 192.168.20.11 อยู่ใน VLAN นี้

ดังนั้น, VXLAN frame ที่มี VNI 5020 น่าจะเป็น packet ที่ encapsulate traffic ของ VLAN 20 เพื่อให้สามารถส่งข้ามเครือข่าย Layer 3 ผ่าน VXLAN Tunnel ระหว่าง Leaf switches

## 3. จาก Topology และ Wireshark Packet Capture, VXLAN ทำงานอย่างไรระหว่าง Leaf Switches?

- Host H2 (192.168.20.10) ส่ง Packet ไปหา H6 (192.168.20.11)
- Leaf switch ที่เชื่อมต่อกับ H2 ตรวจสอบว่าเป้าหมายอยู่ที่ Leaf อื่น และต้องส่งผ่าน VXLAN Tunnel.
- Leaf switch encapsulate Layer 2 frame เป็น VXLAN packet โดยใส่ VNI = 5020 และส่งไปยัง VTEP ของ Leaf อีกตัวผ่าน Spine
- เมื่อ Leaf switch ปลายทางได้รับ VXLAN packet มันจะ decapsulate และส่งไปยัง H6 ใน VLAN 20

## 4. MAC Address ที่ปรากฏใน VXLAN Frame ของ Wireshark Capture บ่งบอกถึงอะไร?

Source MAC: เป็น MAC ของ Host ที่ส่ง

Destination MAC: เป็น MAC ของ Host ปลายทาง

แต่ใน VXLAN header จะมี outer Ethernet header ซึ่งใช้ MAC ของ Leaf switches แทน

Outer Source MAC: เป็น MAC ของ Leaf switch ที่ encapsulate packet

Outer Destination MAC: เป็น MAC ของ Leaf switch ปลายทางที่ต้องรับ packet



Capturing from Standard input

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

icmp

No.	Time	Source	Destination	Protocol	Length	Info
11	12.608173	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0
12	12.614021	192.168.20.11	192.168.20.10	ICMP	148	Echo (ping) reply id=0
13	13.619293	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0
14	13.624770	192.168.20.11	192.168.20.10	ICMP	148	Echo (ping) reply id=0
15	14.629037	192.168.20.10	192.168.20.11	ICMP	148	Echo (ping) request id=0

> Frame 11: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface -, id 0

> Ethernet II, Src: 50:00:00:02:00:07 (50:00:00:02:00:07), Dst: 50:00:00:01:00:07 (50:00:00:01:00:07)

> Internet Protocol Version 4, Src: 10.0.0.1, Dst: 10.0.0.4

> User Datagram Protocol, Src Port: 63392, Dst Port: 4789

Virtual eXtensible Local Area Network

> Flags: 0x0800, VXLAN Network ID (VNI)

Group Policy ID: 0

VXLAN Network Identifier (VNI): 5020

Reserved: 0

Ethernet II, Src: Private\_66:68:07 (00:50:79:66:68:07), Dst: Private\_66:68:06 (00:50:79:66:68:06)

> Destination: Private\_66:68:06 (00:50:79:66:68:06)

> Source: Private\_66:68:07 (00:50:79:66:68:07)

Type: IPv4 (0x0800)

[Stream index: 6]

> Internet Protocol Version 4, Src: 192.168.20.10, Dst: 192.168.20.11

> Internet Control Message Protocol

H2C

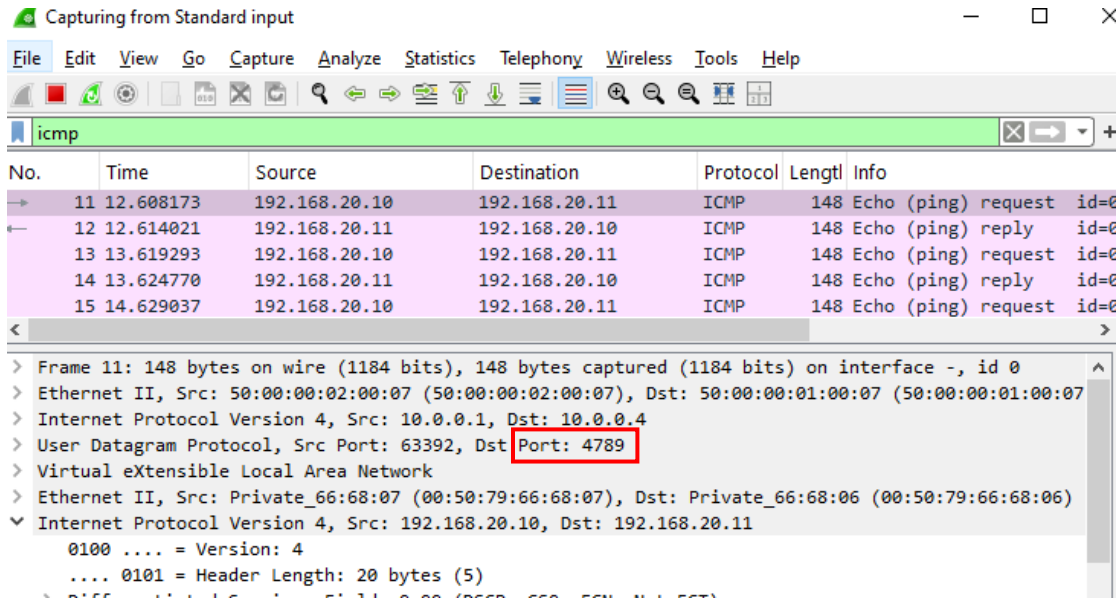
```
GATEWAY      : 192.168.20.1
DNS           :
MAC          : 00:50:79:66:68:07
LPORT        : 20000
RHOST:PORT    : 127.0.0.1:30000
MTU          : 1500
```

##### 5. VXLAN ใช้ UDP Port 4789 ใน Wireshark Packet หมายถึงอะไร?

UDP Port 4789 เป็นค่า port ที่ IANA กำหนดให้ VXLAN ใช้เป็นค่าเริ่มต้นสำหรับ encapsulation

Packet ที่ใช้ UDP Port 4789 แสดงว่าเป็น VXLAN Encapsulated Packet

ใน Capture, Frame ที่มี UDP Port 4789 หมายความว่า เป็น VXLAN traffic ที่ถูก encapsulate และกำลังถูกส่งระหว่าง Leaf switches



#### 6. ในกรณีนี้ VXLAN Packet ถูกส่งจาก 10.0.0.4 ไปยัง 10.0.0.1 แสดงถึงอะไร?

จาก Topology:

10.0.0.1 และ 10.0.0.4 เป็น Loopback interfaces ของ Leaf switches

Loopback นี้ถูกใช้เป็น **VTEP (VXLAN Tunnel End Point)**

เมื่อ Wireshark แสดงว่า VXLAN packet ถูกส่งจาก 10.0.0.4 ไปยัง 10.0.0.1, หมายความว่า Leaf ที่มี 10.0.0.4 เป็น VTEP ได้ encapsulate packet และส่งไปยัง Leaf ที่มี 10.0.0.1 เป็น VTEP ปลายทางผ่าน Spine

```
leaf4# show ip int br

IP Interface Status for VRF "default"(1)
Interface      IP Address      Interface Status
Lo0            10.0.0.4        protocol-up/link-up/admin-up
Eth1/1         10.1.4.1        protocol-up/link-up/admin-up
leaf4#
```

```
leaf1# show ip int br

IP Interface Status for VRF "default"(1)
Interface      IP Address      Interface Status
Lo0            10.0.0.1        protocol-up/link-up/admin-up
Eth1/1         10.1.1.1        protocol-up/link-up/admin-up
leaf1#
```

## 7. VXLAN Packet ใน Capture นี้สามารถระบุได้ว่ามีการใช้ Multicast หรือ Unicast สำหรับการ Forward Packet หรือไม่?

ใน VXLAN, การส่ง packet ไปยัง Leaf อื่นสามารถใช้ได้ 2 วิธีหลัก:

- Multicast VXLAN – ใช้ Multicast Group เพื่อส่ง Broadcast/Unknown/Multicast (BUM) traffic ไปยัง Leaf หลายตัว
- Unicast VXLAN (EVPN) – ใช้ BGP EVPN ในการทำ MAC Address Learning

จาก Wireshark Capture:

มีการใช้ Multicast Address 224.0.0.13 และ 224.0.0.5 ซึ่งเกี่ยวข้องกับ OSPF และ PIM

ไม่มีการใช้ VXLAN Multicast Address เช่น 239.x.x.x

ดังนั้น, ในกรณีนี้ VXLAN อาจใช้ Unicast EVPN มากกว่าการใช้ Multicast VXLAN.

## 8. มีข้อสังเกตใดเกี่ยวกับ OSPF Hello Packet ใน Wireshark Capture และความสัมพันธ์กับ VXLAN?

Wireshark Capture แสดง OSPF Hello Packet ที่ถูกแลกเปลี่ยนระหว่าง 10.1.1.1, 10.1.1.2, และ 10.1.1.3, ซึ่งเป็นการสร้าง Neighbor Relationship ระหว่าง Spine และ Leaf ผ่าน Interface P2P

ความสัมพันธ์ของ OSPF กับ VXLAN:

OSPF ถูกใช้สำหรับการสร้าง Underlay Network เพื่อให้ Spine และ Leaf รู้จักเส้นทางไปยัง VTEP อื่น

เมื่อ OSPF ทำงาน, เส้นทางของ VTEP (10.0.0.1, 10.0.0.4) ถูกแจกจ่ายให้ทุก Leaf รู้จัก

เมื่อมี VXLAN traffic, Leaf switches จะรู้ว่า VTEP ปลายทางอยู่ที่ไหน และสามารถส่ง VXLAN Encapsulated Packet ไปยัง Leaf อื่นได้

## Conclusion

Lab นี้มุ่งเน้นการทำงานของ VXLAN (Virtual eXtensible LAN) ซึ่งเป็นเทคโนโลยีที่ช่วยขยายเครือข่าย Layer 2 ผ่าน Layer 3 โดยใช้เทคนิค encapsulation ผ่าน UDP ทำให้สามารถรองรับเครือข่ายเสมือนได้มากถึง 16 ล้านเครือข่าย เทียบกับ VLAN ที่รองรับเพียง 4096 เครือข่าย โครงสร้างของ VXLAN อาศัยอุปกรณ์ที่เรียกว่า

VTEP (VXLAN Tunnel End Point) ทำหน้าที่ encapsulate และ decapsulate VXLAN packet เพื่อให้สามารถส่งข้อมูลข้ามเครือข่าย Layer 3 ได้อย่างมีประสิทธิภาพ

จาก topology ในการทดสอบนี้ มีการใช้ VXLAN เชื่อมต่อ VLAN 10 และ VLAN 20 ระหว่างอุปกรณ์ Leaf switches โดยที่ VLAN 20 ถูกกำหนดให้ใช้ VXLAN Network Identifier (VNI) 5020 ในการตรวจสอบ Wireshark พบว่ามี packet ที่ encapsulate อยู่ใน VXLAN และใช้ UDP Port 4789 ซึ่งเป็นพอร์ตมาตรฐานของ VXLAN frame การส่งข้อมูลเกิดขึ้นเมื่อ host 192.168.20.10 ต้องการสื่อสารกับ 192.168.20.11 ซึ่งอยู่บน Leaf switch อื่น Leaf switch ต้นทาง encapsulate frame ของ VLAN 20 เป็น VXLAN packet และส่งไปยัง Leaf switch ปลายทางผ่าน Spine switch เมื่อ Leaf switch ปลายทางได้รับ packet จะทำการ decapsulate และส่งต่อไปให้ host ปลายทางใน VLAN 20 ได้อย่างถูกต้อง

อีกหนึ่งประเด็นสำคัญในแล็บนี้คือ การทำงานของ OSPF ที่ทำหน้าที่เป็น Underlay Network โดยช่วยให้ Leaf และ Spine switches สามารถแลกเปลี่ยนเส้นทางและค้นหา VTEP ปลายทางได้อย่างถูกต้อง Wireshark แสดงให้เห็นว่า OSPF Hello Packet ถูกแลกเปลี่ยนระหว่าง Spine และ Leaf เพื่อสร้าง Neighbor Relationship และแจกจ่ายเส้นทางของ VTEP ผ่าน OSPF Routing Table นอกจากนี้ การใช้ Multicast Address 224.0.0.13 และ 224.0.0.5 บ่งชี้ว่ามีการใช้ PIM และ OSPF เพื่อรองรับการทำ VXLAN ในรูปแบบ Unicast EVPN มากกว่าการใช้ Multicast VXLAN

แล็บนี้มีประโยชน์อย่างมากในการทำความเข้าใจการทำงานของ VXLAN และการนำไปใช้จริงใน Data Center Network ซึ่งช่วยลดข้อจำกัดของ VLAN ดั้งเดิมและช่วยให้สามารถสร้างเครือข่ายที่ยืดหยุ่นมากขึ้น นอกจากนี้ยังเป็นการศึกษาแนวทางการออกแบบ Overlay Network และ Underlay Network เพื่อให้ระบบเครือข่ายสามารถทำงานได้อย่างเสถียรและรองรับการขยายตัวของระบบได้ในอนาคต