



UNIVERSITA' DEGLI STUDI DI  
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base  
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Cloud And Network  
Infrastructures

# *VXLAN in Topologia Leaf-Spine Layer 3*

Anno Accademico 2022/2023

Andrea Olandese  
matr. M63001304

Adriano Coppola  
matr. M63001303

# Contents

<b>1</b>	<b>ContainerLab e tecnologie ausiliarie</b>	<b>1</b>
1.1	Container . . . . .	2
1.2	VXLAN . . . . .	2
<b>2</b>	<b>Topologia di rete</b>	<b>4</b>
<b>3</b>	<b>Dimostrazione pratica</b>	<b>7</b>

# Chapter 1

## ContainerLab e tecnologie ausiliarie

**ContainerLab** è un emulatore di rete open-source che costruisce scenari di rete, chiamati *labs*, configurati in file **YAML**. Questi permettono di descrivere la topologia della rete specificando i *Network Operating Systems* da containerizzare, i loro collegamenti, i volumi da montare, le interfacce di amministrazione e i comandi da eseguire all'avvio dei container. Permette di raggiungere i container risolvendo il nome tramite il file */etc/hosts* e di mostrare grafi della topologia tramite web o file *dot*.

L'obiettivo del progetto è implementare un *overlay Layer 2 VXLAN* composto da tre host su un *underlay Layer 3* composto da una topologia *Leaf-Spine*. Di seguito sono illustrate le tecnologie utilizzate.

## 1.1 Container

I router Leaf e Spine sono istanze **FRR** (*Free Range Routing*). Questo è una fork di **Quagga**, un pacchetto software che fornisce una suite di protocolli di routing sia statico che dinamico. È composto dal daemon **zebra** e dai daemon di routing. Può essere configurato tramite **vysh**, una *cli modale* stile Cisco.

I protocolli utilizzati per il routing sono **OSPF** e **BGP**. Il primo è utilizzato per il routing dei pacchetti, mentre il secondo serve ad effettuare il tunneling **evpn** verso gli *Autonomous Systems* della topologia.

## 1.2 VXLAN

*Virtual eXtensible LAN* è una tecnologia di virtualizzazione di rete che ha come obiettivo principale l'abbattimento del vincolo di 4096 VLAN in contesto *multitenancy*. Incapsula i pacchetti in un tunnel UDP con porta destinazione 4789. Il pacchetto interno può contenere un VLAN ID scelto dal tenant. L'incapsulamento e il decapsulamento avvengono ai **VTEPs** (*VXLAN Tunnel End Points*).

VXLAN implementa il tunneling evpn nei router Leaf, che assumono il ruolo di VTEPs. Questo avviene aggiungendo tre interfacce:

- br0: interfaccia di bridging
- vxlan10: interfaccia che implementa VXLAN con VXLAN ID 10

- eth3: interfaccia collegata all'host.

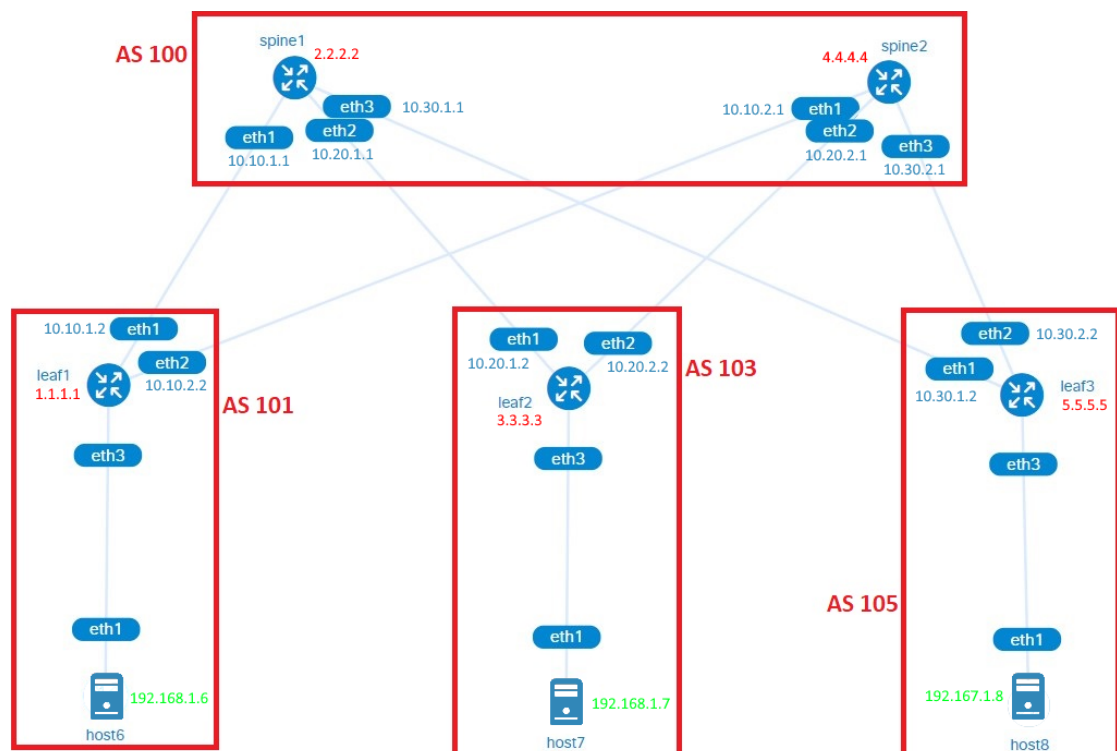
L'interfaccia br0 fa da master per le altre due. I pacchetti che arrivano all'interfaccia eth3 e destinati ad un indirizzo IP che fa parte della rete overlay sono inviati al bridge, che riflette il pacchetto all'interfaccia vxlan10. A questo punto, il kernel incapsula il pacchetto e lo invia sull'interfaccia adatta. Quando viene ricevuto un pacchetto con indirizzo sorgente appartenente all'overlay il flusso di dati è inverso ma entrano in gioco gli stessi attori.

La scoperta degli altri VTEPs e le rotte per raggiungerli è effettuata da BGP, che quindi popola la *VTEP Layer 2 table*.

## Chapter 2

# Topologia di rete

Ai fini di creare una rete che sfrutti le tecnologie descritte, è stata implementata una rete Leaf-Spine che implementa OSPF come protocollo L3 e usa BGP-EVPN per creare un tunnel VXLAN L2 tra gli host.



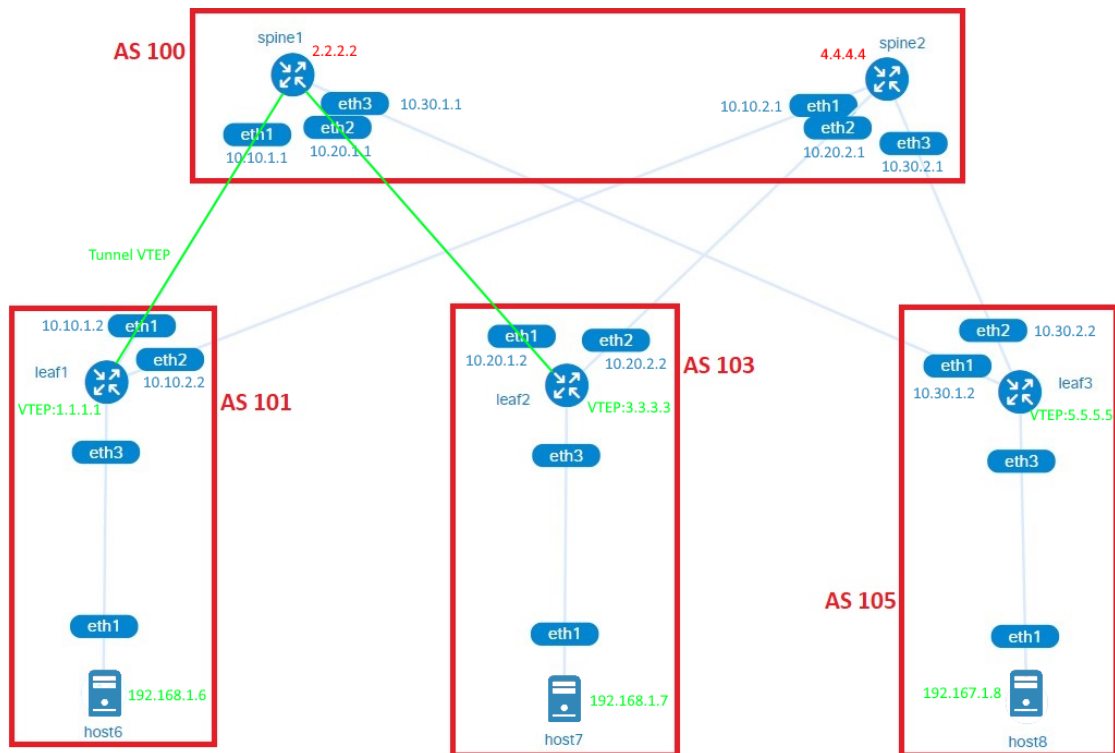
La rete sopra illustrata è composta da 3 router Leaf, a ognuno dei quali è collegato un host, e 2 router Spine, i quali sono collegati ad ogni singolo Leaf.

I Leaf router sono collegati a livello 2 con gli host e a livello 3 con gli Spine, mediante il protocollo OSPF.

Ogni coppia Leaf-host è stata considerata come nel medesimo Autonomous System, separato dalle altre, mentre gli Spine sono stati considerati nello stesso AS.

La topologia mostra come ogni host faccia parte della medesima sottorete, essendo collegati tramite la tecnologia VXLAN.

I Leaf Switch fanno da VTEP per creare tunnel con gli altri Leaf, permettendo agli host di comunicare come fossero ad un hop di distanza.





## Chapter 3

# Dimostrazione pratica

Per verificare la presenza di connettività nella rete, ci colleghiamo all'host 6 e facciamo un Ping verso l'host 7.

```
| 5 | clab-vxlan-leaf2 | 22b94ed79162 | frrouting/frr
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
64 bytes from 192.168.1.7: icmp_seq=1 ttl=64 time=0.148 ms
64 bytes from 192.168.1.7: icmp_seq=2 ttl=64 time=0.120 ms
64 bytes from 192.168.1.7: icmp_seq=3 ttl=64 time=0.093 ms
64 bytes from 192.168.1.7: icmp_seq=4 ttl=64 time=0.099 ms
64 bytes from 192.168.1.7: icmp_seq=5 ttl=64 time=0.101 ms
^C
--- 192.168.1.7 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4128ms
rtt min/avg/max/mdev = 0.093/0.112/0.148/0.020 ms
```

Verificata la connettività, si può notare ora come vengono inviati i pacchetti di ICMP Echo Request.

Mettendosi in ascolto sull'interfaccia eth1 del secondo Leaf, si può notare come questi pacchetti ICMP siano incapsulati in VXLAN, che vede come sorgente e destinazione i due leaf che fanno da VTEP per il tunnel tra gli host.

```

Frame 134: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface 0
Ethernet II, Src: aa:c1:ab:88:ca:f5 (aa:c1:ab:88:ca:f5), Dst: aa:c1:ab:ab:da:31 (aa:c1:ab:ab:da:31)
Internet Protocol Version 4, Src: 10.10.1.2, Dst: 10.20.1.2
User Datagram Protocol, Src Port: 45879, Dst Port: 4789
Virtual eXtensible Local Area Network
  Flags: 0x0800, VXLAN Network ID (VNI)
    0... .. = GBP Extension: Not defined
    .... .0.. .. = Don't Learn: False
    .... 1... .. = VXLAN Network ID (VNI): True
    .... .. 0... = Policy Applied: False
    .000 .000 0.00 .000 = Reserved(R): 0x0000
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 10
  Reserved: 0
Ethernet II, Src: aa:c1:ab:88:ef:1f (aa:c1:ab:88:ef:1f), Dst: aa:c1:ab:25:11:14 (aa:c1:ab:25:11:14)
Internet Protocol Version 4, Src: 192.168.1.6, Dst: 192.168.1.7
Internet Control Message Protocol

Frame 135: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface 0
Ethernet II, Src: aa:c1:ab:ab:da:31 (aa:c1:ab:ab:da:31), Dst: aa:c1:ab:88:ca:f5 (aa:c1:ab:88:ca:f5)
Internet Protocol Version 4, Src: 10.20.1.2, Dst: 10.10.1.2
User Datagram Protocol, Src Port: 45879, Dst Port: 4789
Virtual eXtensible Local Area Network
  Flags: 0x0800, VXLAN Network ID (VNI)
    0... .. = GBP Extension: Not defined
    .... .0.. .. = Don't Learn: False
    .... 1... .. = VXLAN Network ID (VNI): True
    .... .. 0... = Policy Applied: False
    .000 .000 0.00 .000 = Reserved(R): 0x0000
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 10
  Reserved: 0
Ethernet II, Src: aa:c1:ab:25:11:14 (aa:c1:ab:25:11:14), Dst: aa:c1:ab:88:ef:1f (aa:c1:ab:88:ef:1f)
Internet Protocol Version 4, Src: 192.168.1.7, Dst: 192.168.1.6
Internet Control Message Protocol

```

Si può notare dalle interfacce utilizzate dai router Leaf che è stato usato lo Spine 1 per inviarsi i pacchetti VXLAN.

Allo scopo di verificare il funzionamento del secondo Spine come percorso alternativo, è stato ripetuto l'esperimento, buttando prima giù l'interfaccia eth1 del secondo Leaf.

```

/ # ping 192.168.1.7
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
64 bytes from 192.168.1.7: icmp_seq=10 ttl=64 time=1.41 ms
64 bytes from 192.168.1.7: icmp_seq=11 ttl=64 time=0.123 ms
64 bytes from 192.168.1.7: icmp_seq=12 ttl=64 time=0.087 ms
64 bytes from 192.168.1.7: icmp_seq=13 ttl=64 time=0.085 ms
64 bytes from 192.168.1.7: icmp_seq=14 ttl=64 time=0.098 ms

```

L'interfaccia buttata giù non appartiene al router al quale è collegato l'host che esegue il Ping, ma dopo un periodo di assestamento, una nuova rotta viene utilizzata per portare i pacchetti VXLAN.

Mettendosi in ascolto sull'interfaccia eth2 del secondo Leaf, si ha la conferma che viene sfruttato il secondo Spine.

```

Frame 59: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface 0
Ethernet II, Src: aa:c1:ab:ea:23:f5 (aa:c1:ab:ea:23:f5), Dst: aa:c1:ab:fa:fb:ad (aa:c1:ab:fa:fb:ad)
Internet Protocol Version 4, Src: 10.10.2.2, Dst: 10.20.2.2
User Datagram Protocol, Src Port: 45879, Dst Port: 4789
Virtual eXtensible Local Area Network
  Flags: 0x0800, VXLAN Network ID (VNI)
    0... .. = GBP Extension: Not defined
    ... ..0.. .. = Don't Learn: False
    ... 1... .. = VXLAN Network ID (VNI): True
    ... ..0... .. = Policy Applied: False
    .000 .000 0.00 .000 = Reserved(R): 0x0000
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 10
  Reserved: 0
Ethernet II, Src: aa:c1:ab:88:ef:1f (aa:c1:ab:88:ef:1f), Dst: aa:c1:ab:25:11:14 (aa:c1:ab:25:11:14)
Internet Protocol Version 4, Src: 192.168.1.6, Dst: 192.168.1.7
Internet Control Message Protocol

Frame 60: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface 0
Ethernet II, Src: aa:c1:ab:fa:fb:ad (aa:c1:ab:fa:fb:ad), Dst: aa:c1:ab:ea:23:f5 (aa:c1:ab:ea:23:f5)
Internet Protocol Version 4, Src: 10.20.2.2, Dst: 10.10.2.2
User Datagram Protocol, Src Port: 45879, Dst Port: 4789
Virtual eXtensible Local Area Network
  Flags: 0x0800, VXLAN Network ID (VNI)
    0... .. = GBP Extension: Not defined
    ... ..0.. .. = Don't Learn: False
    ... 1... .. = VXLAN Network ID (VNI): True
    ... ..0... .. = Policy Applied: False
    .000 .000 0.00 .000 = Reserved(R): 0x0000
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 10
  Reserved: 0
Ethernet II, Src: aa:c1:ab:25:11:14 (aa:c1:ab:25:11:14), Dst: aa:c1:ab:88:ef:1f (aa:c1:ab:88:ef:1f)
Internet Protocol Version 4, Src: 192.168.1.7, Dst: 192.168.1.6
Internet Control Message Protocol

```

Infine si può verificare come gli host interagiscano come ad un hop di distanza mediante VXLAN, con un comando di Traceroute.

```

/ # traceroute 192.168.1.7
traceroute to 192.168.1.7 (192.168.1.7), 30 hops max, 46 byte packets
 1  192.168.1.7 (192.168.1.7)  0.010 ms  0.007 ms  0.002 ms

```

Come ultima verifica della configurazione, di seguente vi è la tabella BGP del tunneling EVPN, estratta dal router Leaf 1.

```

BGP table version is 6, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
EVPN type-1 prefix: [1]:[EthTag]:[ESI]:[IPlen]:[VTEP-IP]:[Frag-id]
EVPN type-2 prefix: [2]:[EthTag]:[MAClen]:[MAC]:[IPlen]:[IP]
EVPN type-3 prefix: [3]:[EthTag]:[IPlen]:[OrigIP]
EVPN type-4 prefix: [4]:[ESI]:[IPlen]:[OrigIP]
EVPN type-5 prefix: [5]:[EthTag]:[IPlen]:[IP]

      Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 1.1.1.1:2
*> [2]:[0]:[48]:[aa:c1:ab:88:ef:1f]
      1.1.1.1
      ET:8 RT:101:10
      32768 i
*> [3]:[0]:[32]:[1.1.1.1]
      1.1.1.1
      ET:8 RT:101:10
      32768 i
Route Distinguisher: 3.3.3.3:2
*> [2]:[0]:[48]:[aa:c1:ab:25:11:14]
      3.3.3.3
      RT:103:10 ET:8
      0 100 103 i
*> [3]:[0]:[32]:[3.3.3.3]
      3.3.3.3
      RT:103:10 ET:8
      0 100 103 i
Route Distinguisher: 5.5.5.5:2
*> [2]:[0]:[48]:[aa:c1:ab:4d:07:18]
      5.5.5.5
      RT:105:10 ET:8
      0 100 105 i
*
      5.5.5.5
      RT:105:10 ET:8
      0 100 105 i
* [3]:[0]:[32]:[5.5.5.5]
      5.5.5.5
      RT:105:10 ET:8
      0 100 105 i
*>
      5.5.5.5
      RT:105:10 ET:8
      0 100 105 i

```