Contrail Networking Lab Tests_v.4

2021.4.23 HyungKwang Choi

Contrail Networking Lab Tests_v.4

0. Intro & Revision history

This lab test is a series of “Learning Contrail-Networking”, which I posted at LinkedIn.

So, for the chapter, it’s is consecutive from “1. How to install Contrail-Networking v2005 + CentOS7.7”

> Snipping from my post at LinkedIn.

'Contrail-Networking' is OpenStack based Juniper SDN solution for Private Cloud.

i want to share my knowledge below as a series.
Let's start it from there with me, with fun.

* What i am going to post below as a series.

1. How to install Contrail-Networking v2005 + CentOS7.7
2. Lab tests based on 5 scienario easy step by step.
3. How to install RedHat Open Stack13 + contrail v2005
4. Docker network.
5. Finding Contrail vRouter NextHop
6. Common issues and Tips for troubleshooting in Contrail.

> Document Revision History

Revision	Author	Date	Description of Changes
1.0	hkchoi	2/7 2021	Initial draft
2.0	hkchoi	2/17 2021	
3.0	Hkchoi	4/12 2021	
4.0	hkchoi	4/23 2021	Added lab test#5 (Spine & Leaf)

1. Before LAB testing

1.1. Tools useful for debugging.

Below tools useful for troubleshoot. Strongly recommend to install.

1. Install 'tcpdump'
 # yum -y install tcpdump

> tcpdump options
-i : interface
-nn : Doesn't convert/resolve 'protocol' and 'port' to naming.
-e : when printing, it prints link-level header such as) mac address
- w : used to save to a file.
- D : list interfaces for tcpdump

> simple use
tcpdump -i enp5s0f1 ←===recommend to use
tcpdump -**nnei** enp5s0f1
tcpdump -**nnei** enp5s0f1 -w hi.pcap
tcpdump -D

2. "netstat -rn"
#yum -y install net-tools

1.2. MX and EX2200 basic configurations

For lab test#1,#2,#3 and #4, once physical & initial configuration done for underlayer, you don't need to modify that anymore.
key point : for Internal network 100.0.0.0/24, ping must be reachable among nodes, even to the lo0 of MX/QFX.
This is very important. I will mention it later again.

> EX2200 config

```
set system host-name EX2200
set system services ftp
set system services ssh root-login allow
set system services telnet
set system services netconf ssh
set interfaces ge-0/0/0 unit 0 family ethernet-switching port-mode access
set interfaces ge-0/0/0 unit 0 family ethernet-switching vlan members CentOS_contrail
set interfaces ge-0/0/1 unit 0 family ethernet-switching port-mode access
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members CentOS_contrail
set interfaces ge-0/0/4 unit 0 family inet address 100.1.0.1/24
set interfaces ge-0/0/12 unit 0 family ethernet-switching port-mode access
set interfaces ge-0/0/12 unit 0 family ethernet-switching vlan members CentOS_contrail
set interfaces ge-0/0/13 unit 0 family ethernet-switching port-mode access
set interfaces ge-0/0/13 unit 0 family ethernet-switching vlan members CentOS_contrail
set interfaces ge-0/0/23 unit 0 family ethernet-switching port-mode access
set interfaces ge-0/0/23 unit 0 family ethernet-switching vlan members CentOS_contrail
set interfaces ge-0/1/1 unit 0 family inet address 100.2.0.1/24
set interfaces me0 unit 0 family inet address 172.27.122.240/24
```

```
set interfaces vlan unit 100 family inet address 100.0.0.240/24 <====== This is important. It used as virtual-gateway address during your setting.
set routing-options static route 172.27.0.0/16 next-hop 172.27.122.1
set routing-options static route 10.0.0.0/8 next-hop 172.27.122.1
set routing-options static route 3.3.3.71/32 next-hop 100.1.0.2
set routing-options static route 3.3.3.72/32 next-hop 100.2.0.2
set vlans CentOS_contrail vlan-id 100
set vlans CentOS_contrail l3-interface vlan.100
```

> MX960 config

```
set system host-name MX960_spine
set system services ftp connection-limit 10
deactivate system services ftp connection-limit
set system services ssh root-login allow
set system services telnet connection-limit 20
set system services netconf ssh
set system domain-name kornet.net
set system time-zone Asia/Seoul
set system dump-on-panic
set system internet-options no-tcp-reset drop-all-tcp
set system ports console type vt100
set system ports auxiliary type vt100
set system ntp server 172.27.122.253
set chassis fpc 3 pic 0 tunnel-services <== enabling tunnel interface is very important. Please keep in mind that.
set chassis network-services enhanced-ip
set interfaces ge-3/0/0 unit 0 family inet address 100.1.0.2/24
set interfaces xe-3/2/0 unit 0 family inet address 200.0.0.2/24
set interfaces fxp0 unit 0 family inet address 172.27.122.195/24
set interfaces lo0 unit 0 family inet address 3.3.3.71/32
set routing-options static route 3.3.3.72/32 next-hop 100.1.0.1
set routing-options static route 100.0.0.0/24 next-hop 100.1.0.1
set routing-options static route 172.0.0.0/8 next-hop 172.27.122.1
set routing-options static route 10.0.0.0/8 next-hop 172.27.122.1
set routing-options static route 0.0.0.0/0 next-hop 172.27.122.1
set routing-options static route 201.0.0.0/24 next-hop 200.0.0.
```

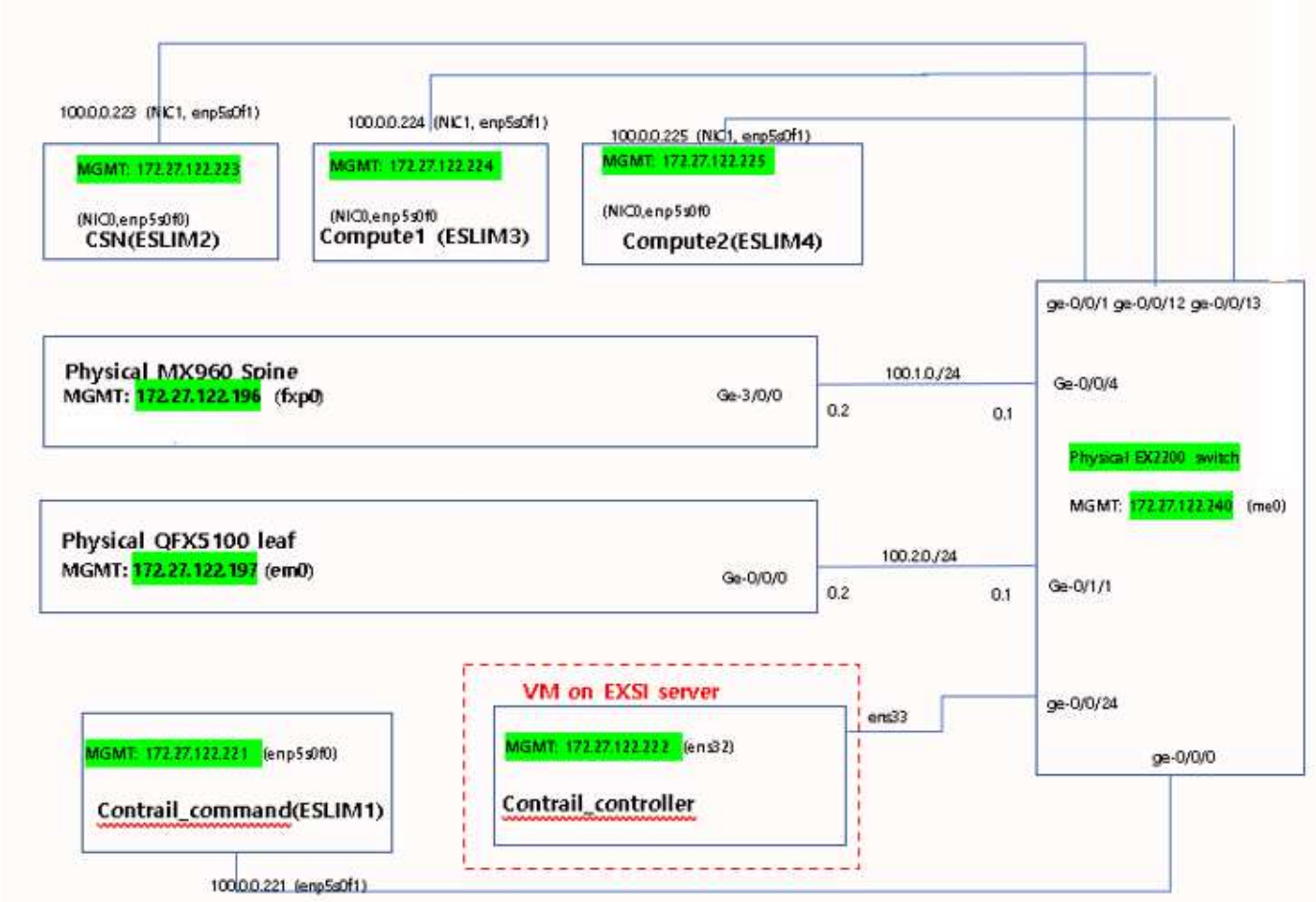
1.3. My lab info

> my lab spec & server naming used.

Device (Host NAME)	Servers	MGMT IP	Internal IP	CPU	RAM	HDD
contrail-command(contrailcommand-221)	Eslim server #1	172.27.122.221 (enp5s0f0)	100.0.0.221(enp5s0f1)	8	16GB	150GB
contrail-Controller (controller-222)	EXSI VM	172.27.122.222 (ens32)	100.0.0.222 (ens33)	8	32GB	100GB
contrail-service-node (CSN-223)	Eslim server #2	172.27.122.223 (enp5s0f0)	100.0..0.223(enp5s0f1)	8	16GB	150GB
Compute 1 (compute1-224)	Eslim server #3	172.27.122.224 (enp5s0f0)	100.0.0.224(enp5s0f1)	8	16GB	150GB
Compute 2 (compute2-225)	Eslim server #4	172.27.122.225 (enp5s0f0)	100.0.0.225(enp5s0f1)	8	16GB	150GB
MX960		172.27.122.195 (fxp0)				
QFX5110		172.27.122.197 (em0)				
EX2200		172.27.122.240 (me0)				

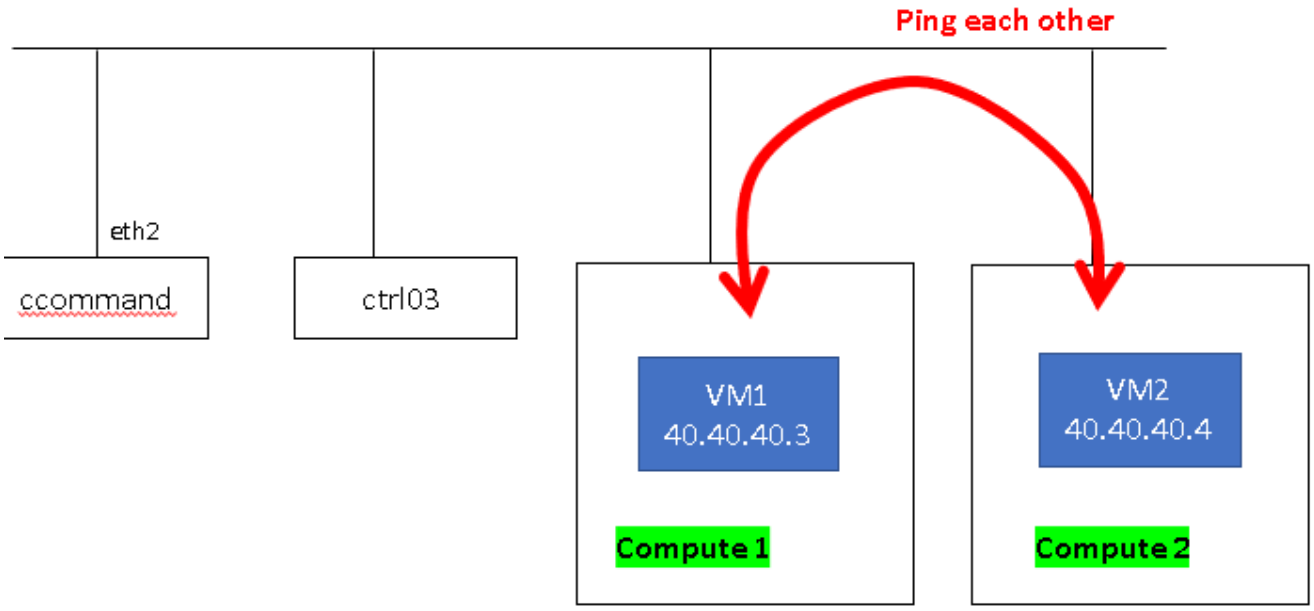
1.4 Basic lab topo

- For contrail installation, at least you need 2 ports.
  => port1 : MGMT , port 2 : internal port for vhost0
- port1 (MGMT) used to install and download overall installation files. So it must be internet/DNS reachable



2. Lab test #1 : Creating 2 VMs and the same VNs (Virtual Network)

> Lab Topo



2.1. image download : cirros for VM image.

> **Download site**: http://download.cirros-cloud.net/0.5.1/

cirros-0.5.1-source.tar.gz	2020-03-06 10:53	542K
cirros-0.5.1-x86_64-disk.img	2020-03-06 11:19	16M
cirros-0.5.1-x86_64-initramfs	2020-03-06 11:18	6.2M
cirros-0.5.1-x86_64-kernel	2020-03-06 11:19	8.7M
cirros-0.5.1-x86_64-lxc.tar.gz	2020-03-06 11:18	5.5M

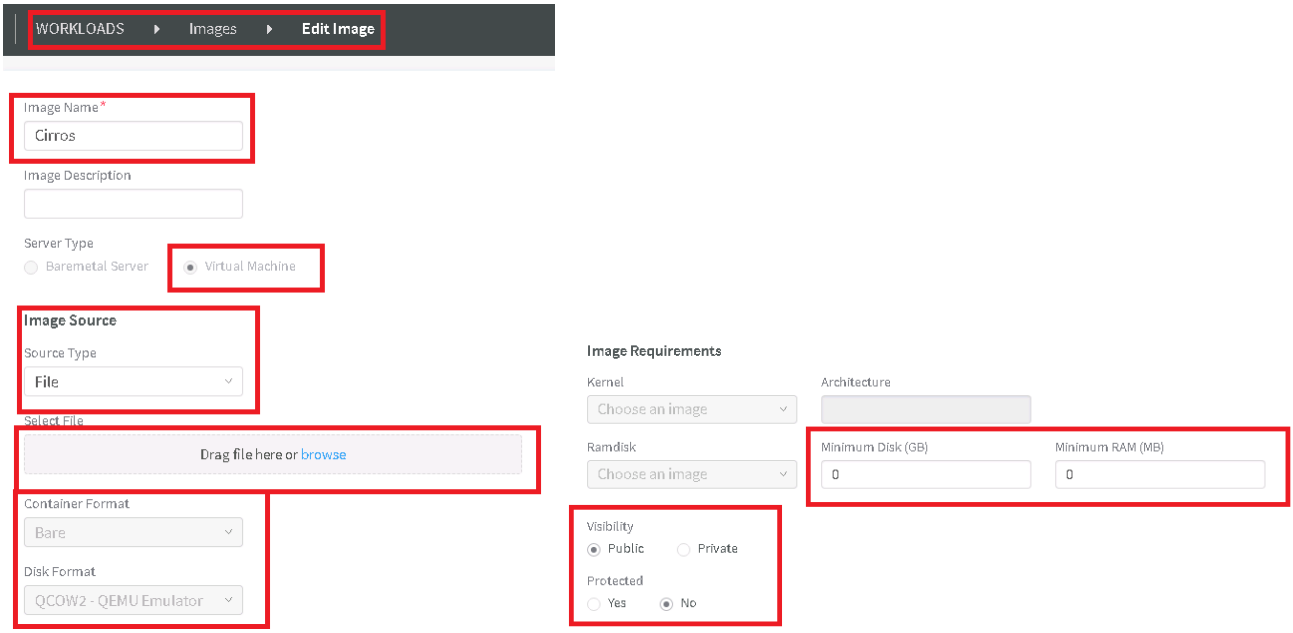
2.2. login to Contrail-Command GUI

**** This Doc is a series of my 'Contrail' knowledge sharing. So to start this chapter, you have to do done the first chapter "How to install Contrail_v2005 + CentOS 7.7". This goes on that.**

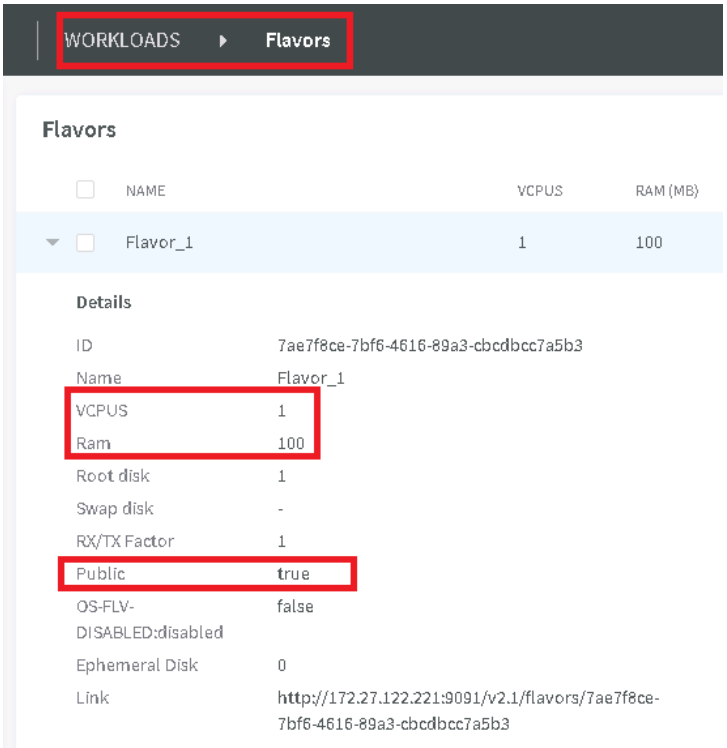
- if you are done 'Contrail' installation, please login to 'contrail-command' node, and all the settings will go on "contrail-command GUI"
- To access 'Contrail-command' server GUI, for web browser you have to use Chrome. Firefox and IE is not compatible.
- GUI : **https://<Contrail-command-server-ip>:9091 ex) https://172.27.122.221:9091**

2.3. Create image.

- VM image which I used “cirros-0.4.0-x86_64-disk.img”.



2.4. Create Flavors.

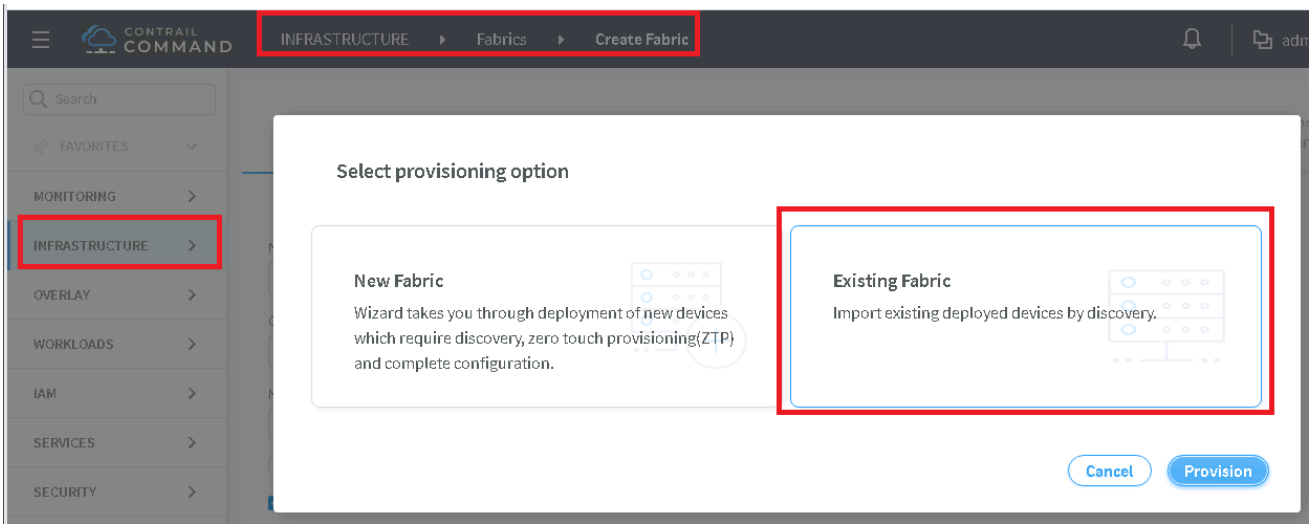


2.5. Create Fabric.

**Tips.
While setting up 'Fabric', sometimes you can face an issue due to several reason.
In this case, for 'Debugging', you can do like that.

ex) In my case, controller IP is '172.27.122.222'. ssh root@172.27.122.222, pw: contrail123
Do login controller server.

[root@controller-222 ~]# tail -f /var/log/contrail/contrail-fabric-ansible-playbooks.log



To remember TERMs (BrownField vs GreenField) *

1. The terms of "BrownField", "GreenField"

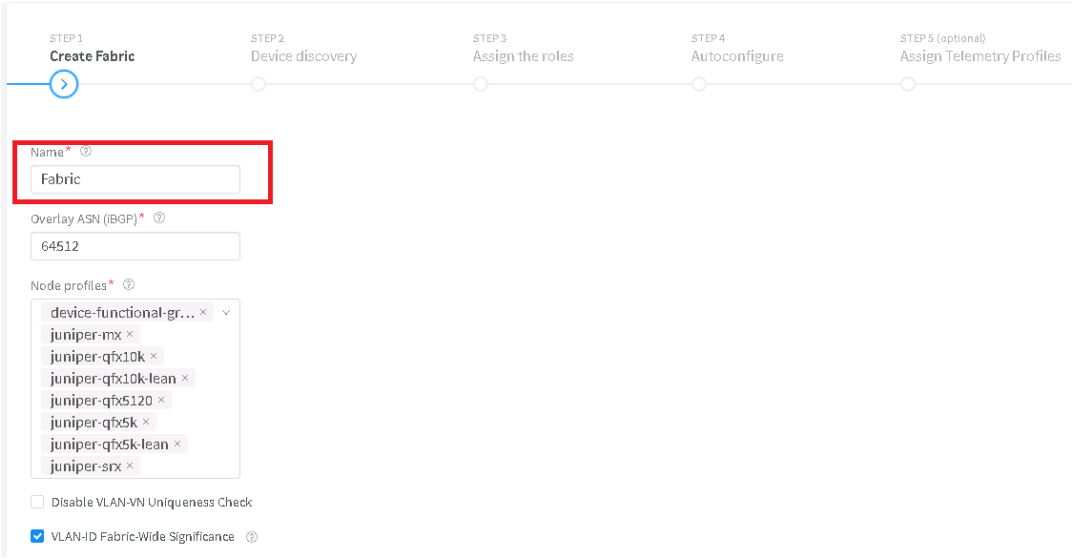
Brownfield devices—You can provision legacy devices or existing devices to form an IP Clos network. Unlike greenfield devices, brownfield devices are manually provisioned before device onboarding. The brownfield fabric workflow includes playbooks that automate the fabric data model creation in the database. You can perform basic device management functions such as image upgrade, device discovery, device underlay configuration, assign roles to devices, and view node profile information.
- operator must set Underlay environment and ping reachable among each contrail node and even lo0 of Router/Switch device.
So contrail does only Overlay related settings.

Greenfield devices—You can provision new devices to form an IP Clos network. These devices are connected to a management network that is provisioned before device onboarding. The greenfield fabric workflow then zero-touch-provisions all factory-default devices to form an operational IP Clos network with underlay connectivity.
This greenfield fabric workflow includes playbooks that automate the fabric data model creation in the database, DHCP server configuration, generating device bootstrap configuration, uploading device bootstrap configuration to TFTP server, device discovery, node profile auto-assignment, device role assignment, and role-based auto configuration.
- It's like a ZTP configuration. Operator doesn't need to set underlay part. Contrail(GreenField) does set up everything for Fabric.

2. Reference
https://www.juniper.net/documentation/en_US/contrail19/topics/concept/ems-fabric-management-overview.html

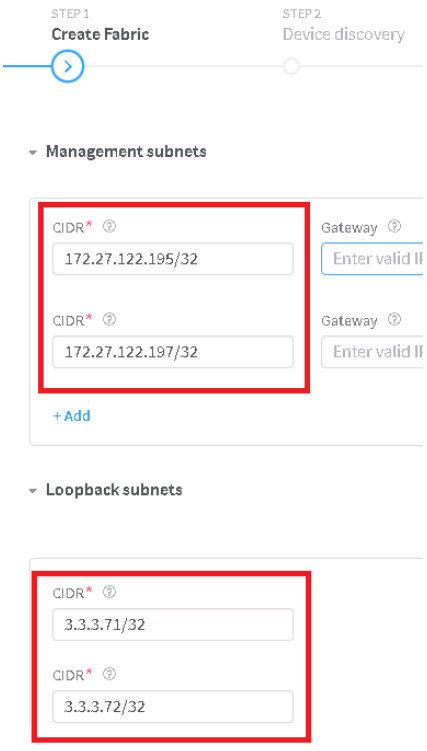
3. YouToube : Brownfeild vs Greenfield

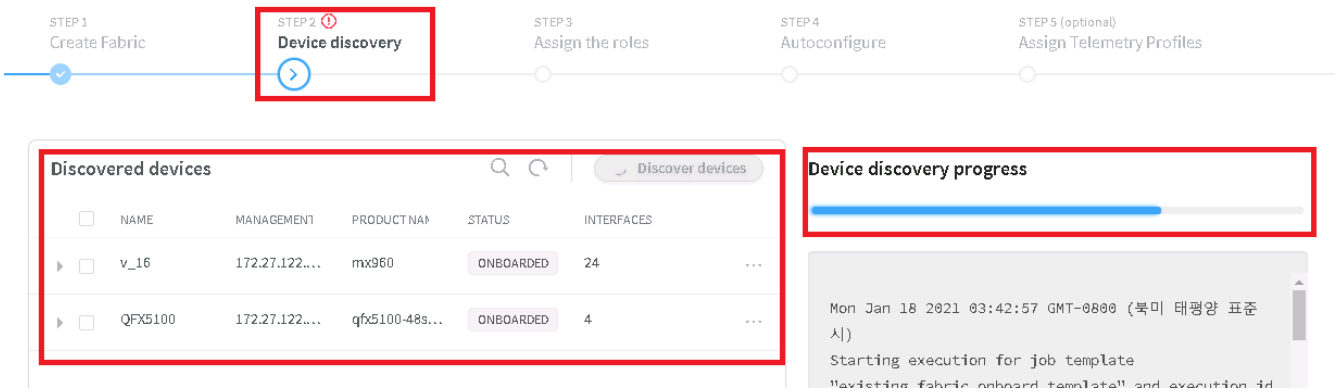
https://www.youtube.com/watch?v=WEFQbzVF6rA



**** Here very important.**

- For Management CIDR, you must put both spine, leaf ip address, and it must be /32 (regardless of it's real subnet)
- Loopback address also must be written with /32
- Of course, you can put /24 instead, but if you do that, sometime you can face an error.

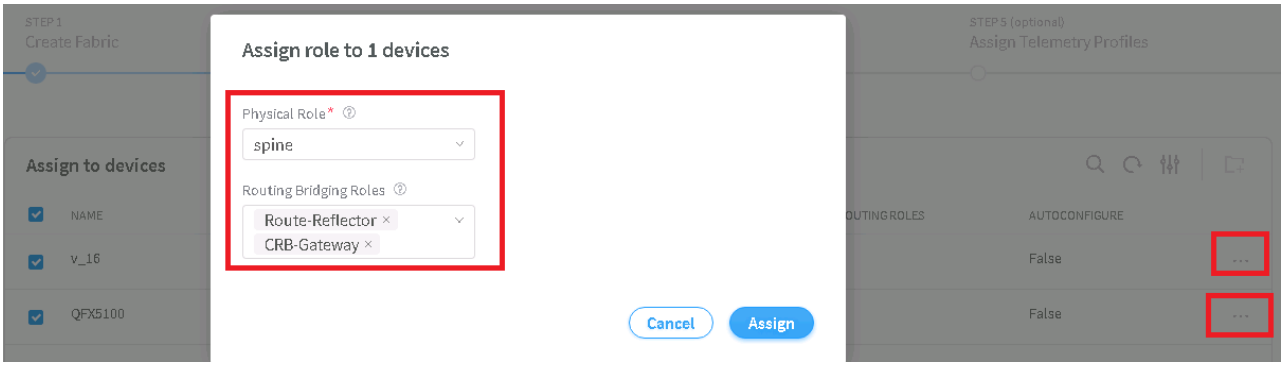


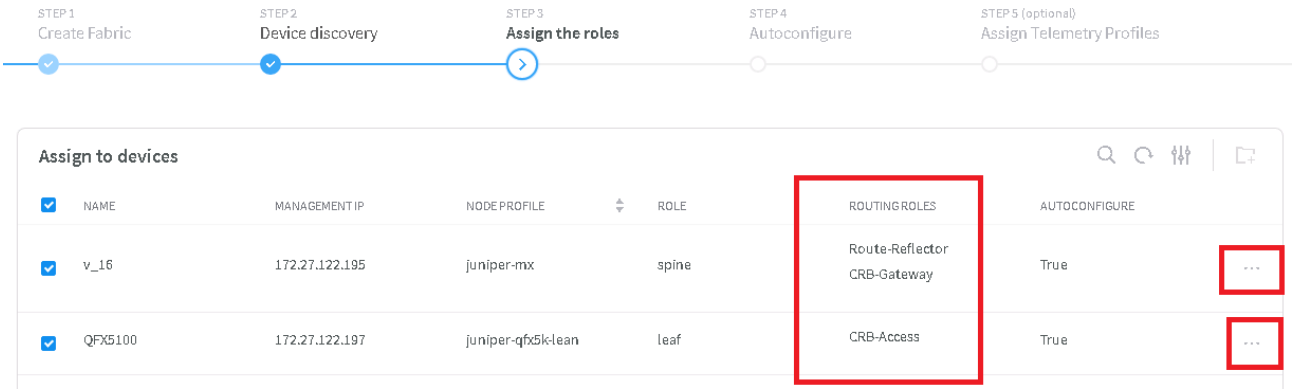


'Role'

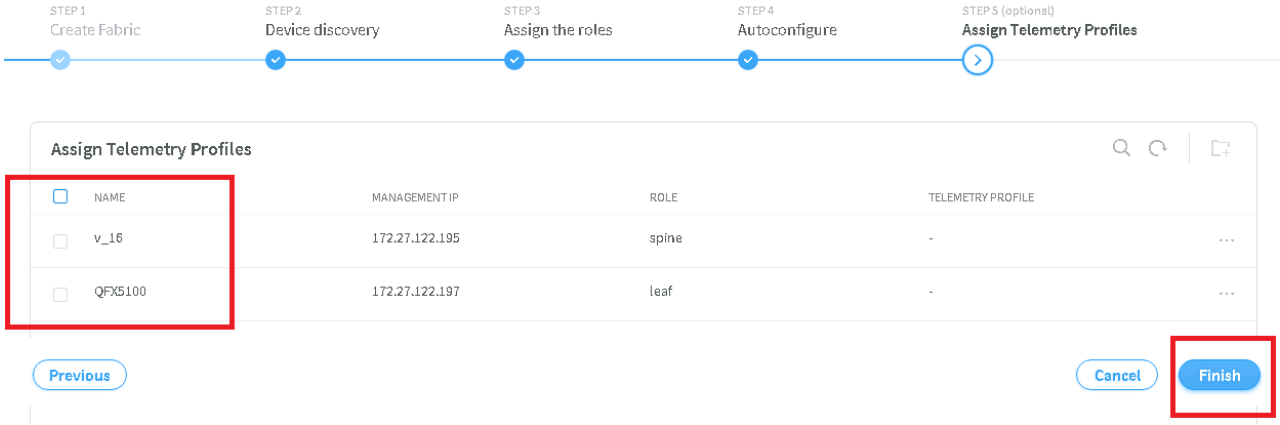
1. CRB-Gateway (CRB-Centrally Routed Bridge)
   - This works as L3 Gateway/Router for internal network among same VN or different VN (virtual Network). That means to use the network internal only. With CRB-Gate Role, you can't send traffic internet (out of contrail network). Because with this role, for L3 Routing to the internet, 0.0.0.0/0 route is not imported.

2. DC-gateway :
   - This is to send traffic to internet. 0.0.0.0/0 default route on inet.0 is imported to VRF routing-instance with EVPN type5. Regarding this, It will be handled at lab test #4.

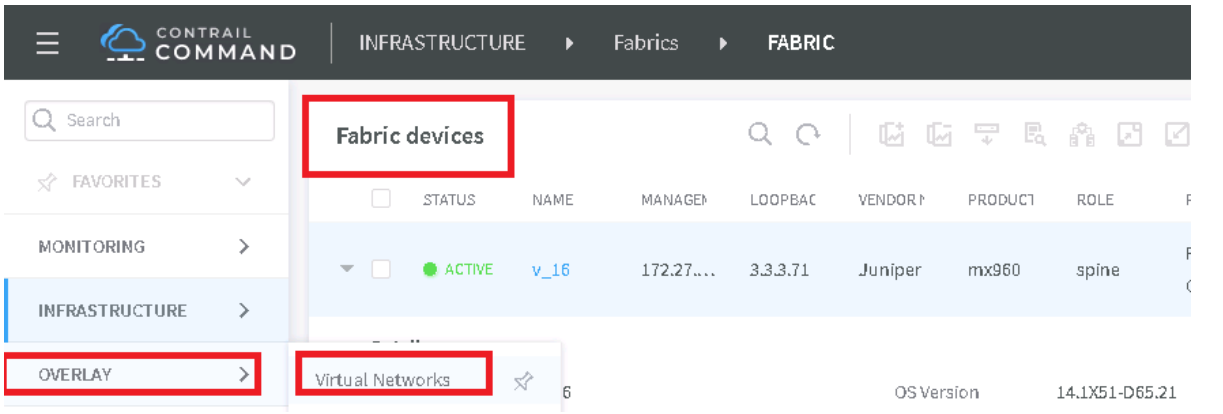


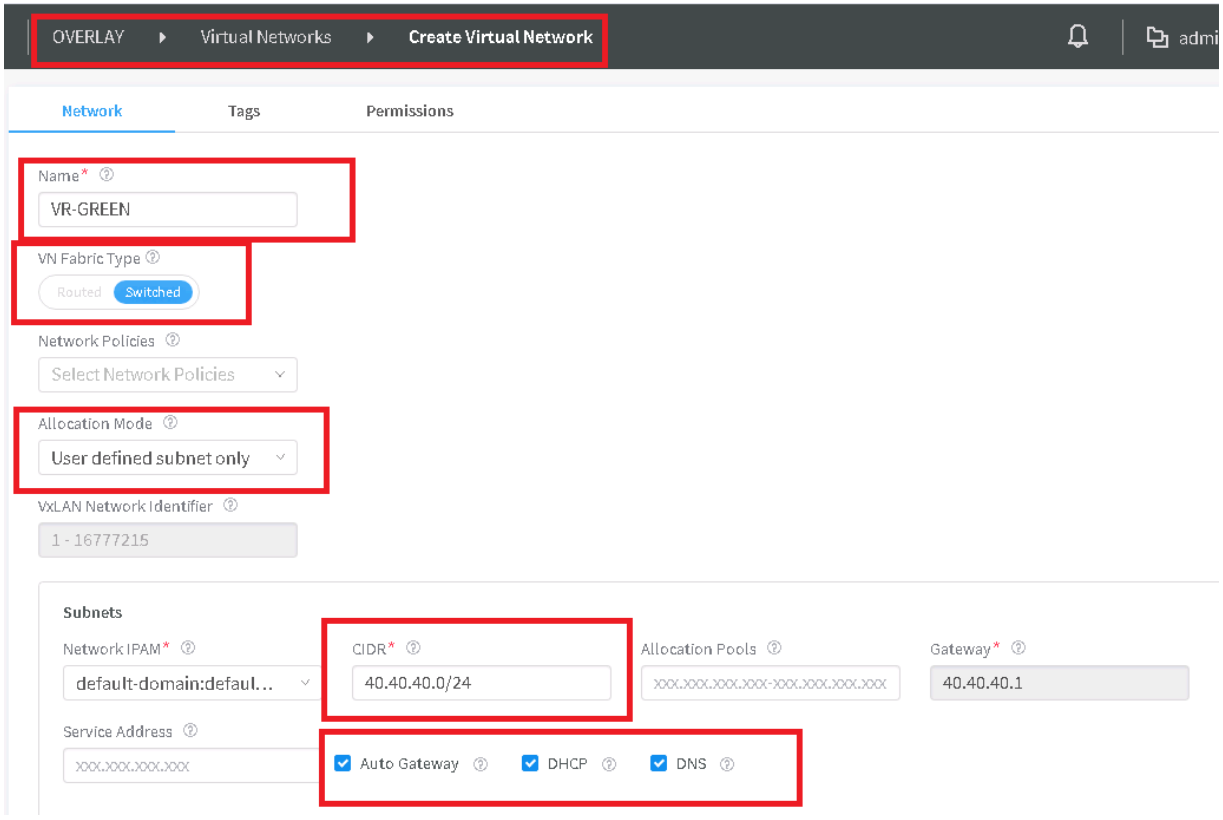


> Telemetry skip

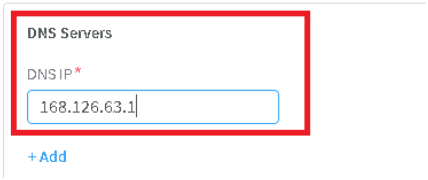


2.6. After 'Fabric' registration done , creating VN (virtual network)

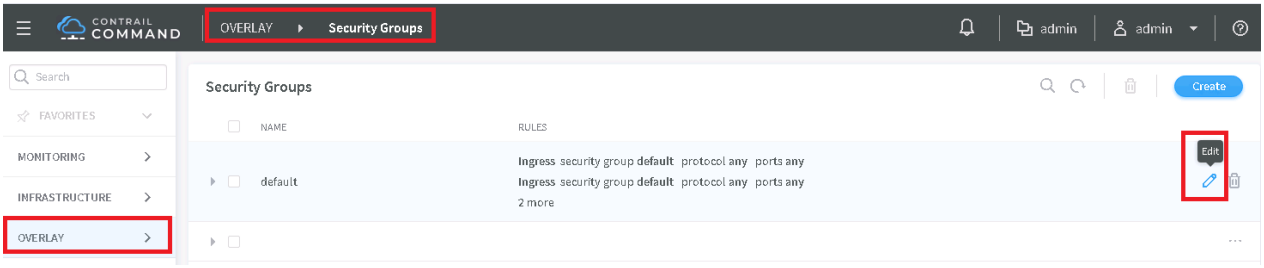




> DNS : if not needed, you can skip

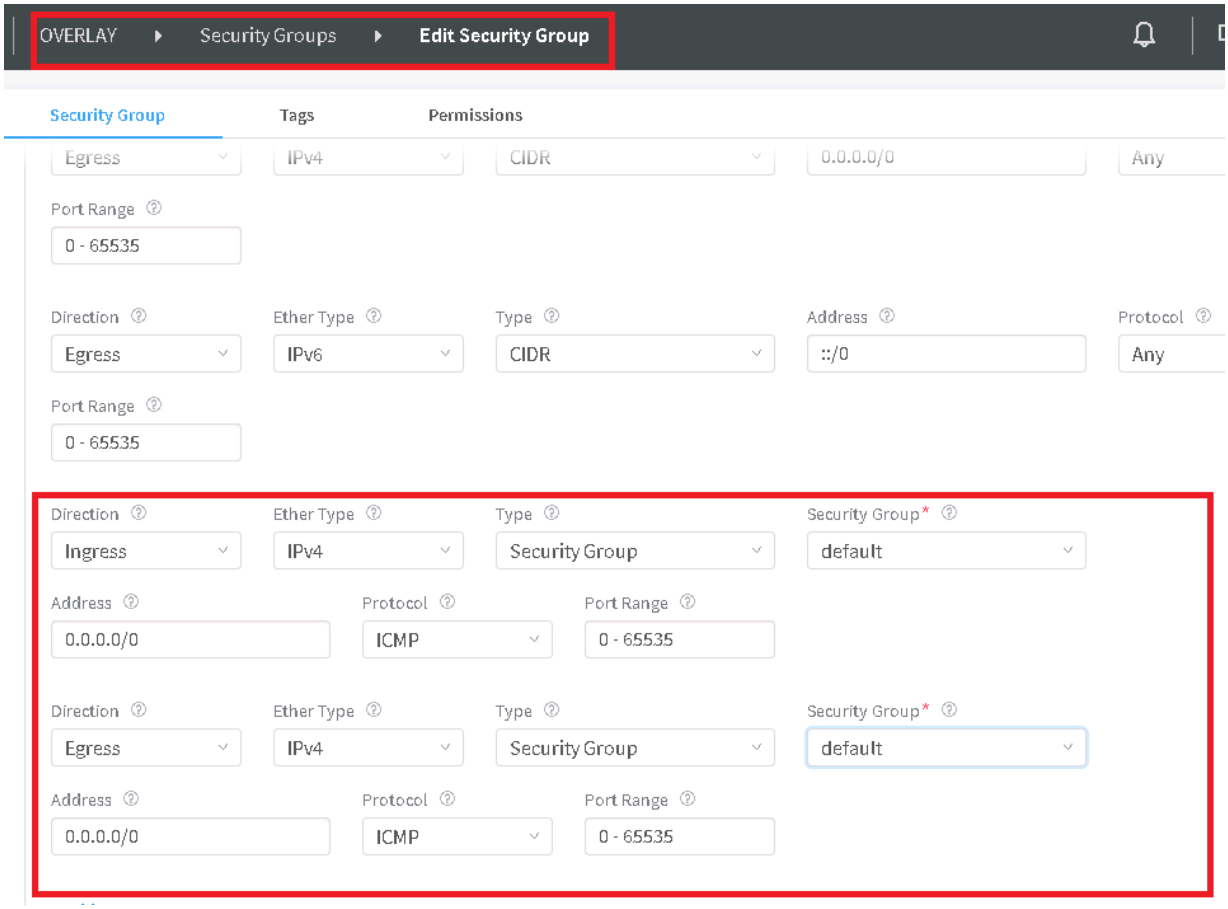


2.7. Create 'Security Group'

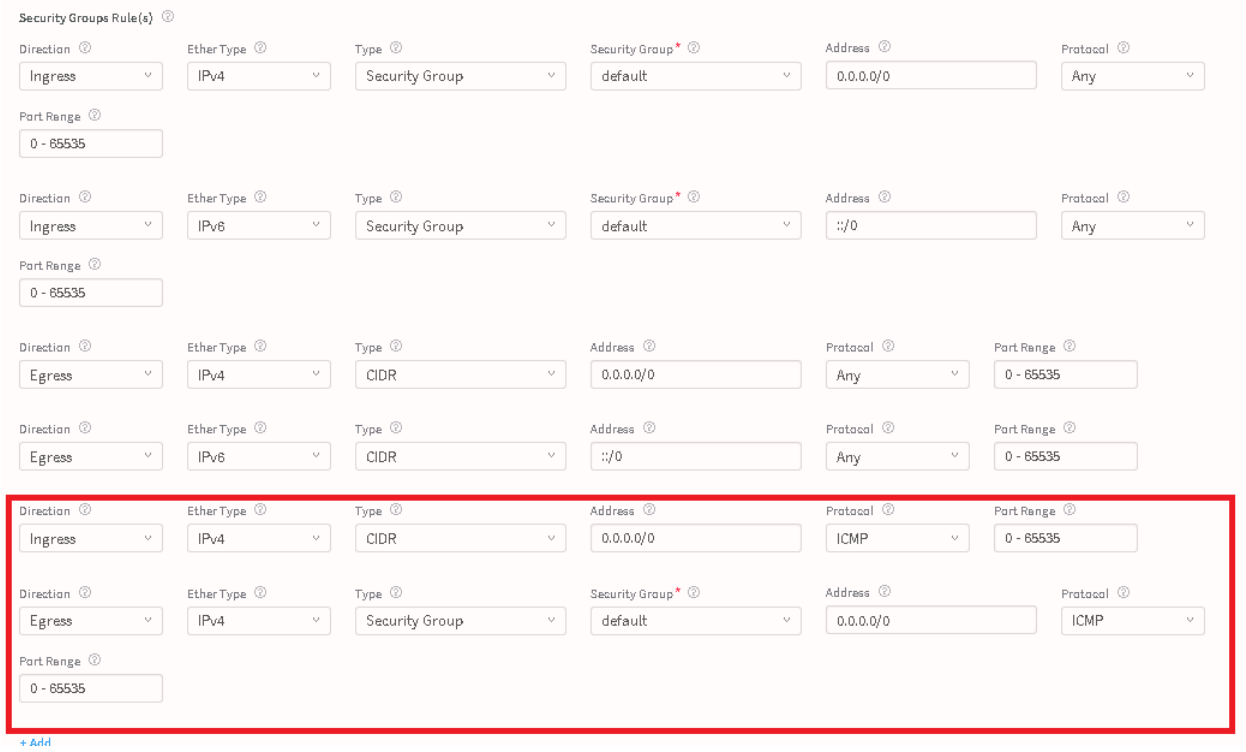


> Please set 'Security Group' like below.

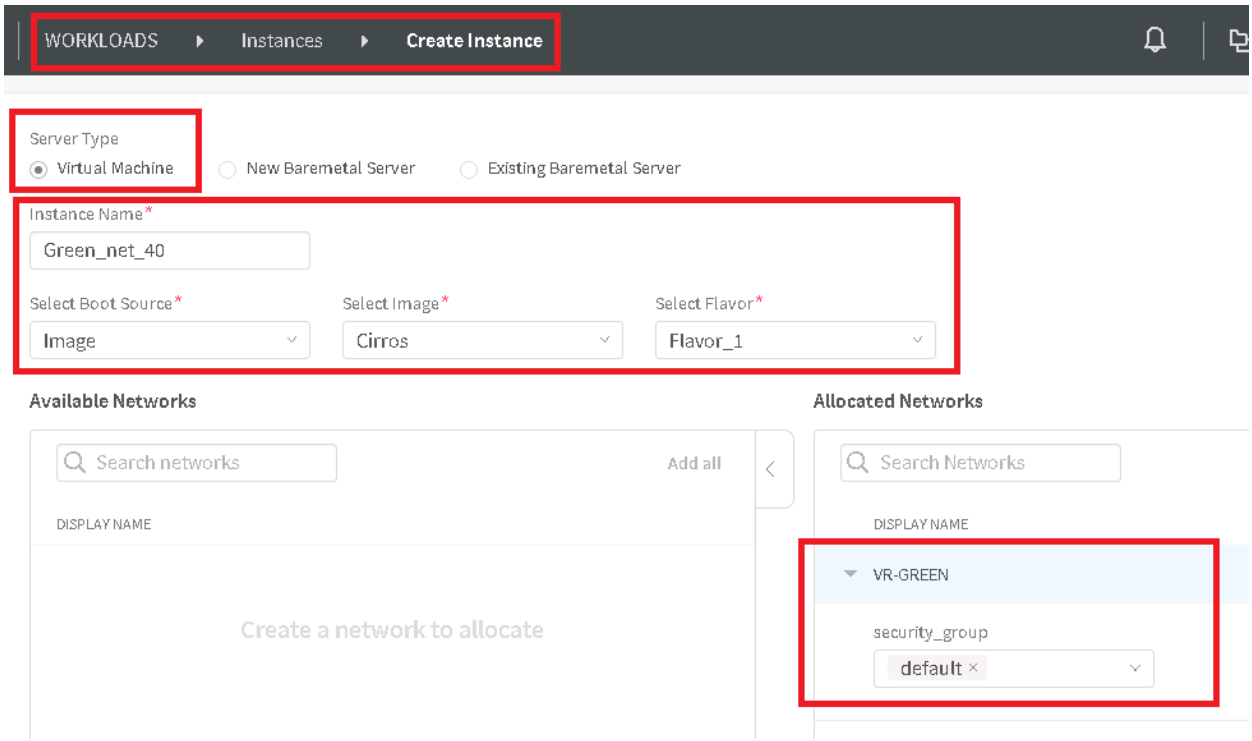
Below parts are wrong. Laster you are going to see, some problem related with ping failure.
So later troubleshooting purpose, please set like below. And remember.



> Actually below parts are correct settings. Later we will check it further.

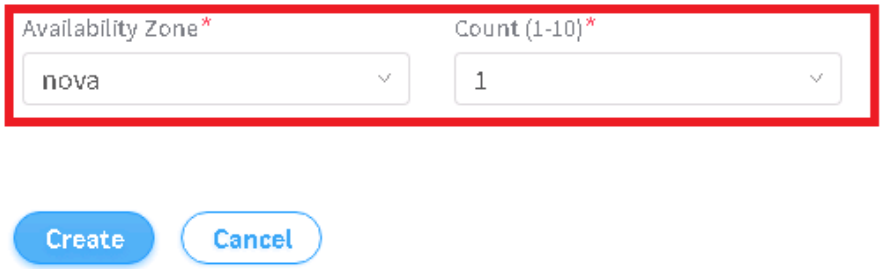


2.8. Create a VM instance, assign VN which you created above.

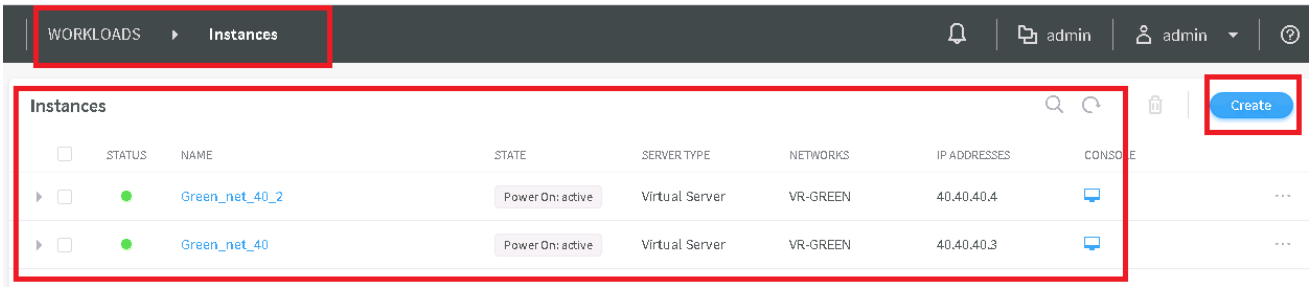


Tips.

Below “Availability Zone”, in drop box, you can see only ‘nova’. That means openstack create a VM instance in random compute node. If you want to point specific compute node to create a VM instance, there is a method.
Login ‘OpenStack’ GUI, there is a option to name each compute node. After naming compute node, then go back to here “contrail-command GUI”, you can see the compute node you named..



2.9. Done creating 2 VMs.



you created a VN (VR-GREEN) and assigned the VN to two VMs.
On the right side, click on the icon of “compute” shape. Then do ping test between VM instances. You can see ping is reachable. That mean, they belong to the same network.

2.10. How to find VM Nex-Hop.

=> This topic will be handled at the next release. **“Finding Contrail vRouter NextHop.pdf”**

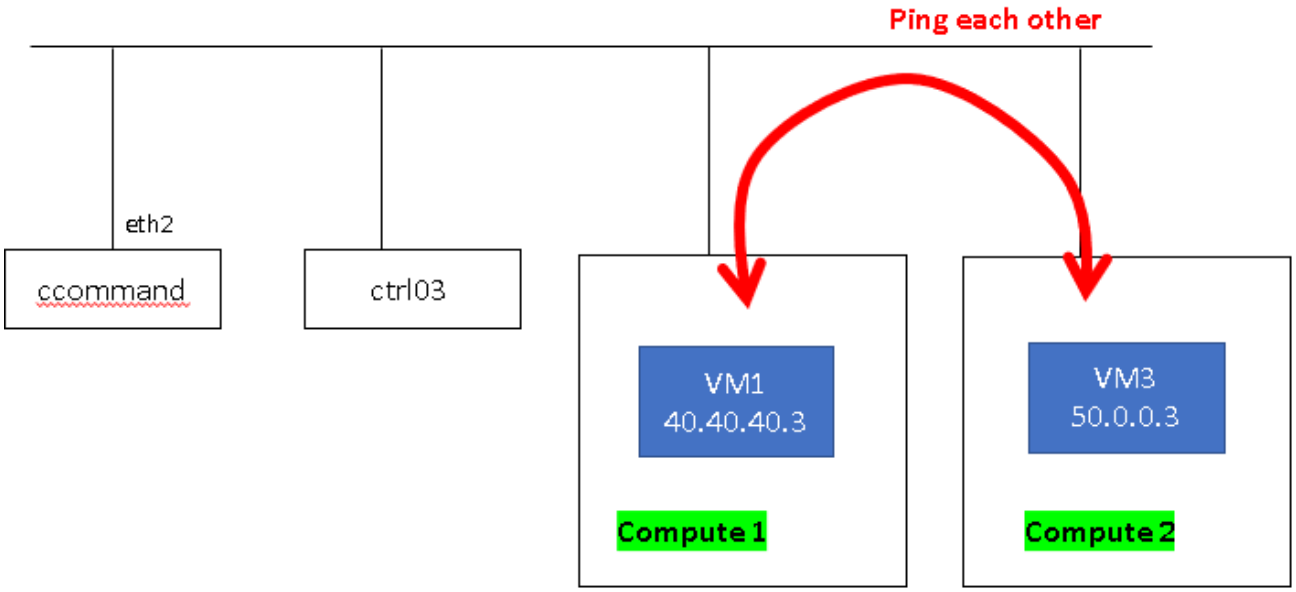
3. Lab test #2 : Creating 2 VMs and different VNs (Virtual Network)

**** Key point : 2 VNs (Virtual-Network) are different network.**

So to communicate different networks, you need a gateway. In contrail, it's called 'logical-router'. In Contrail logical-router function consists of physical router or without physical router (just with contrail controller S/W function).

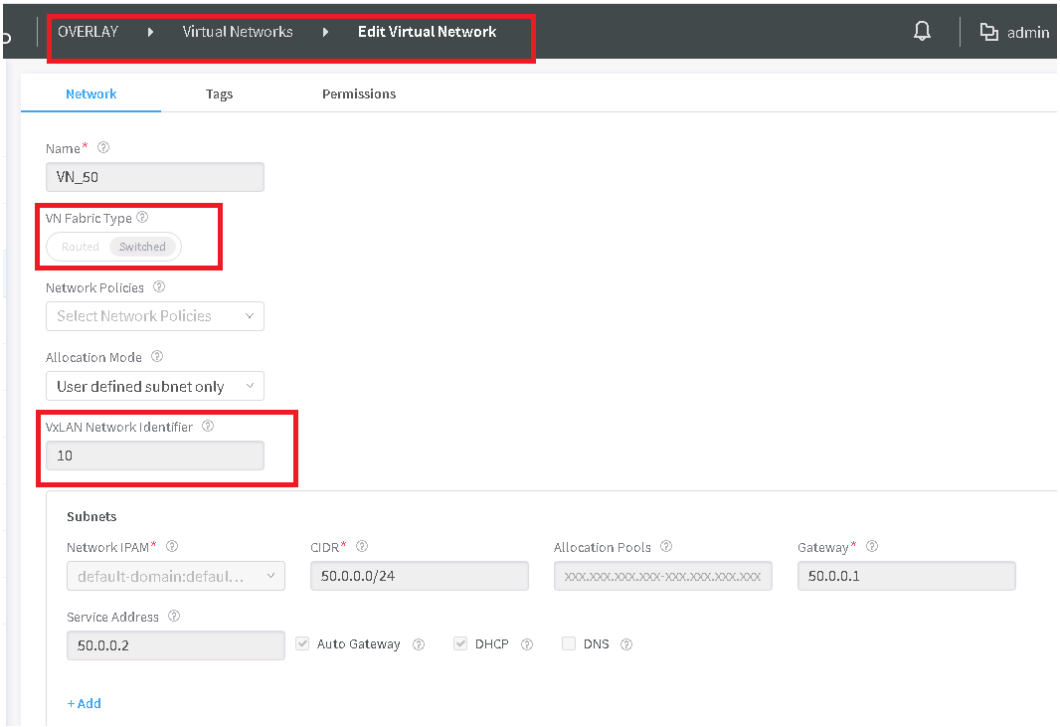
- At this time, we will do create a logical-router without physical router (just with contrail controller S/W function).
- It's easy. Create a new VN(Virtual Network 50.0.0.0/24), then create logical router and assign VNs (VN 40, VN 50) to the logical router.

> Lab Topo

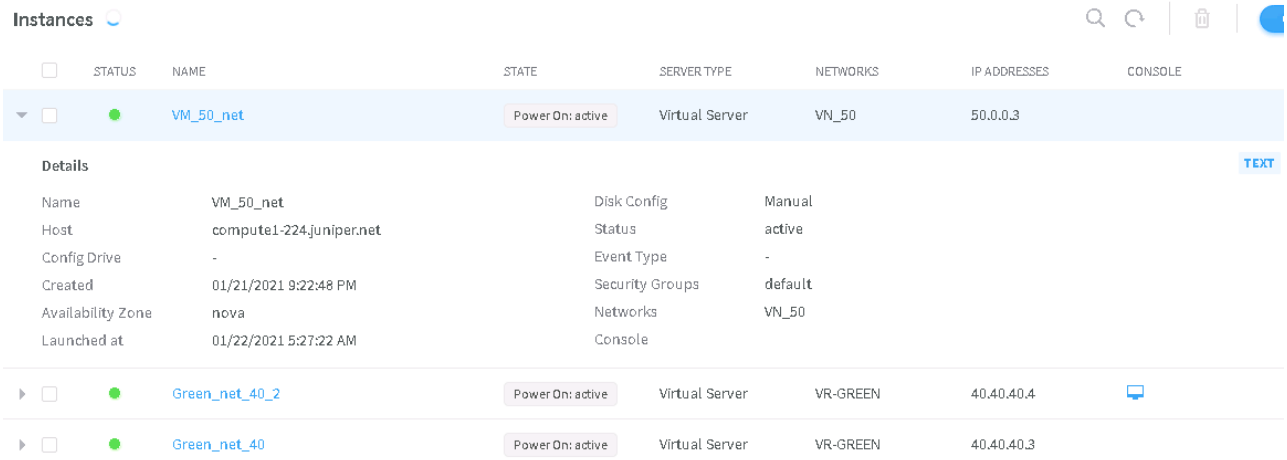


3.1. Creating new VN (virtual Network) : 50-net

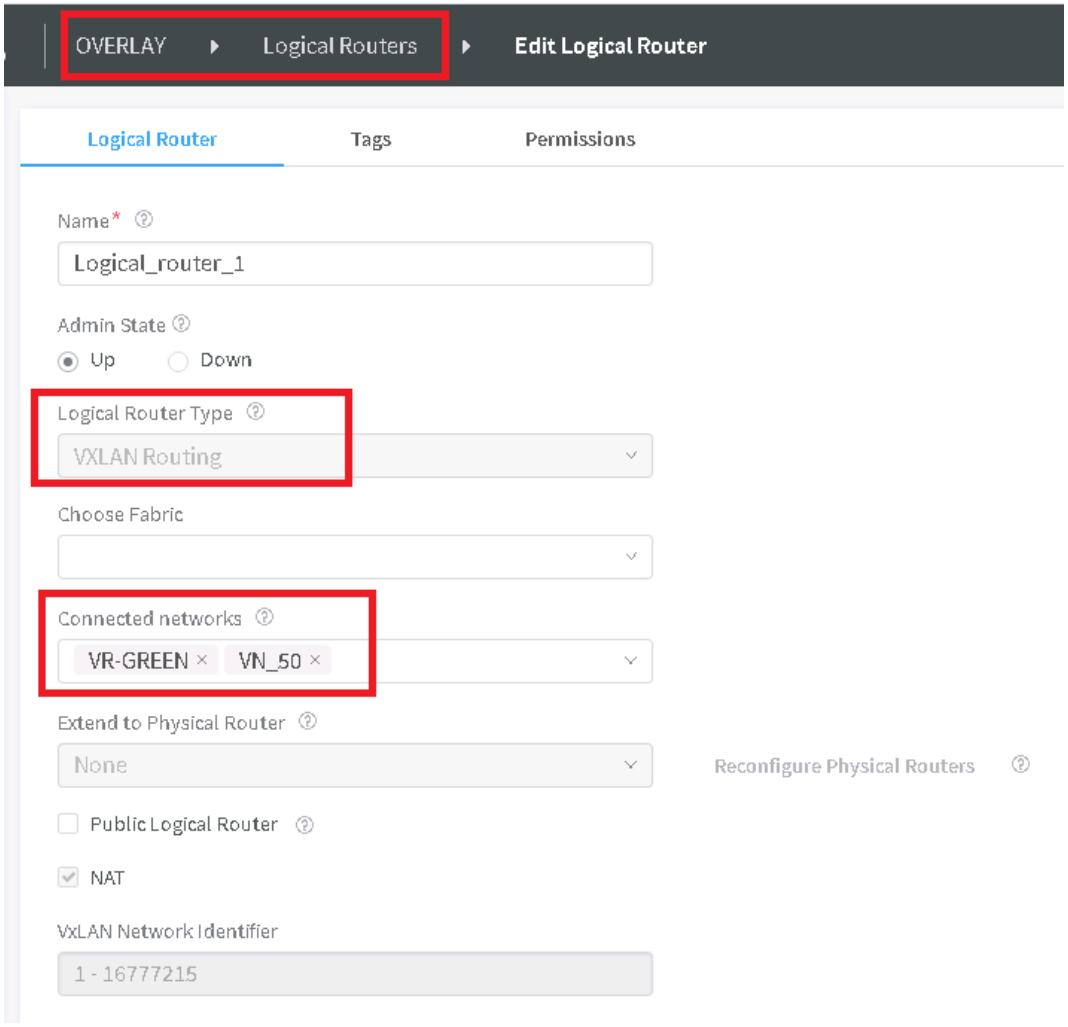
=> After you done creating new VN, the go/do ‘Edit’ mode, you can see VXLAN id automatically created.



3.2. Creating new Instance, and associate VN 50-net



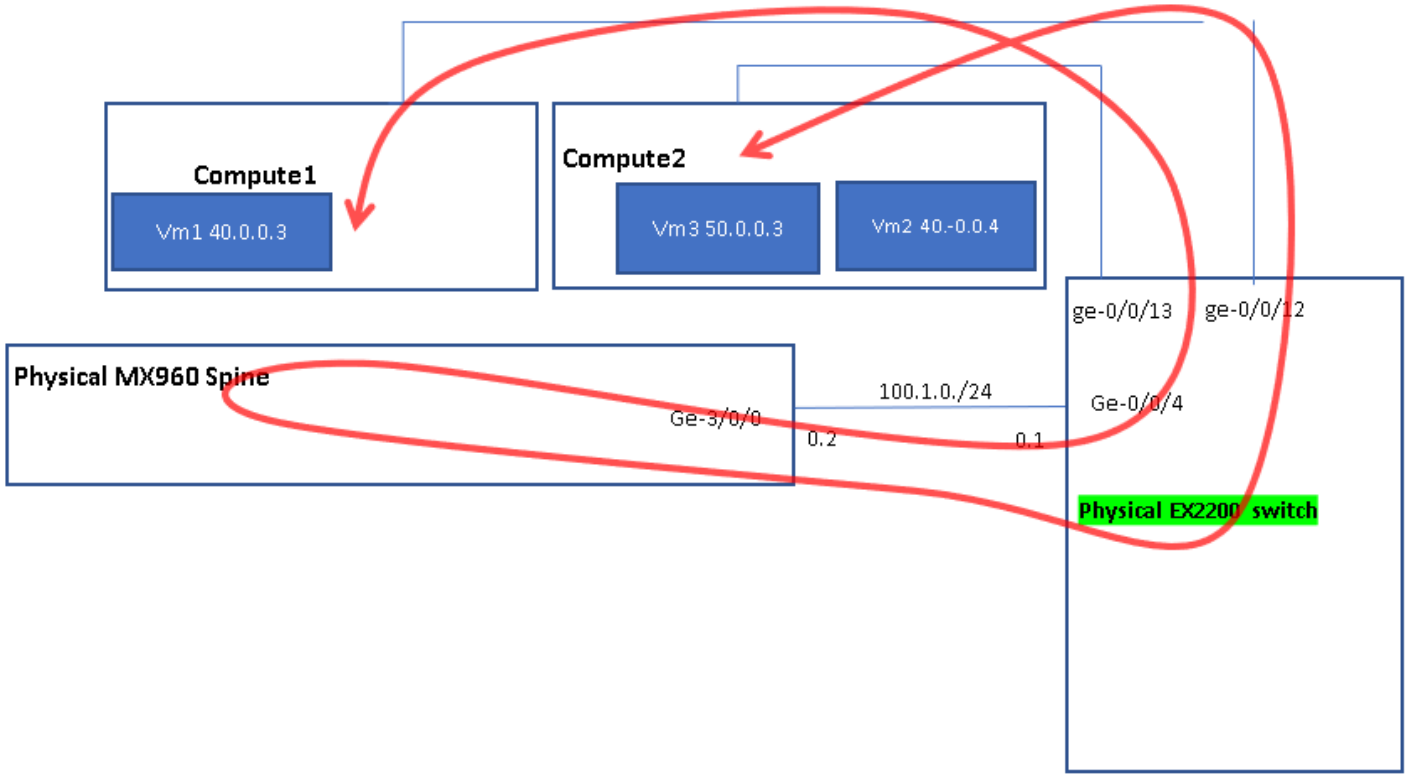
3.3. Creating logical router and put 2 VNs (Net 40 and Net 50)



4. Lab test #3 : Central Routing for different VNs (Virtual Network)

=> implementing 'CRB-Gateway' function. That means with consisting logical-router function with Physical router

> Lab Topo



4.1. Important & be minded for MX vs QFX spine role

When you use MX as Spine for CRB-Gateway or DC-Gateway Role, GRE tunnel interface creation needed, so relevant configuration (" set chassis fpc X pic X tunnel-services") required.
But in CASE QFX, GRE tunnel is not used, but VXLAN tunnel created via physical interface.

* Dynamic-tunnel creation for Spine, below config being pushed by Contrail-Controller
For MX or vMX

```
lab@vmx_re# show | display set | match dyna
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel source-address 1.1.1.1
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel udp
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel destination-networks 192.168.200.0/24
```

* For QFX
GRE tunnel interface not created, but VXLAN tunnel interface created via physical interface

4.2. What CRB-gateway (Central Routed Bridge) is

In Lab#2 we created a logical-router without Physical router.
At this time, we create Fabric with physical router and set physical router to logical-router.
So, for this task, please remove logical-router, which you created Lab#2, and create new logical-router for this chapter.

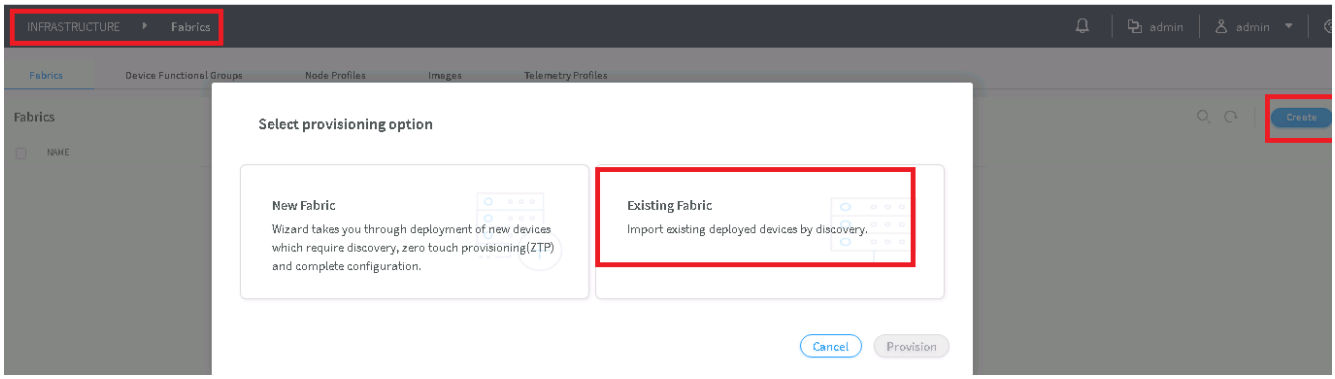
> For this task, When you set up Fabric (Lab#1 Fabric part), for roles, you have to set spine's role as to "Route-Reflector" and "CRB-Gateway". CRB(Centrally Routed Bridge)

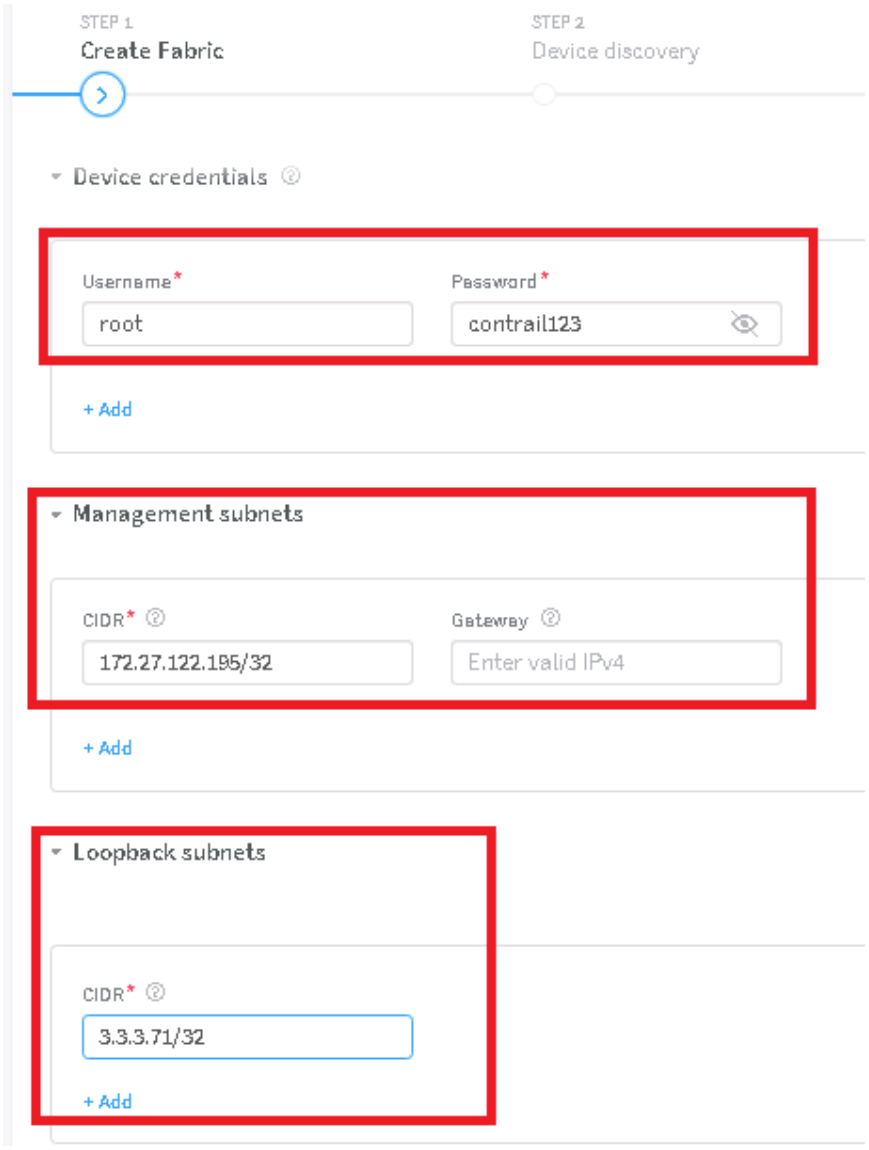
> Please refer to YouTube
Contrail Enterprise Multicloud: Central Routing - YouTube

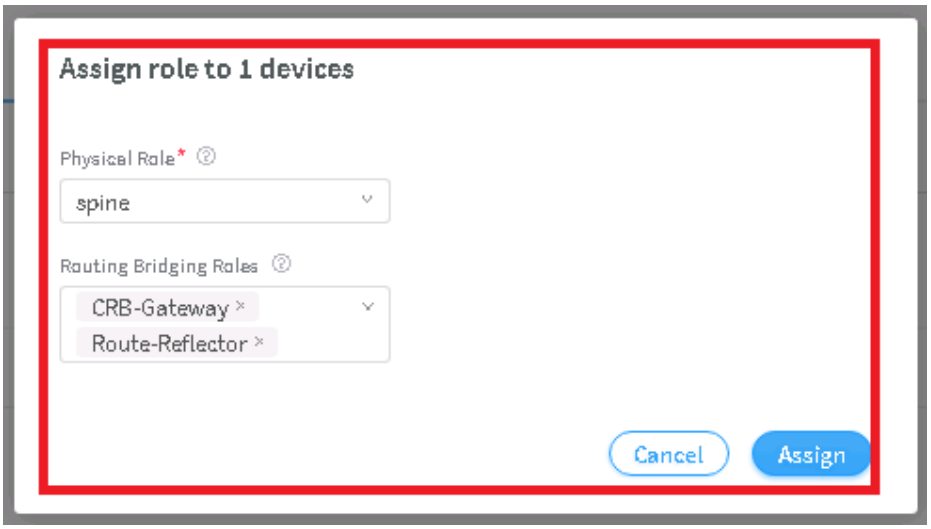
> From juniper web-site.
https://www.juniper.net/documentation/en_US/release-independent/solutions/topics/task/configuration/existing-fabric-cem-configuring.html

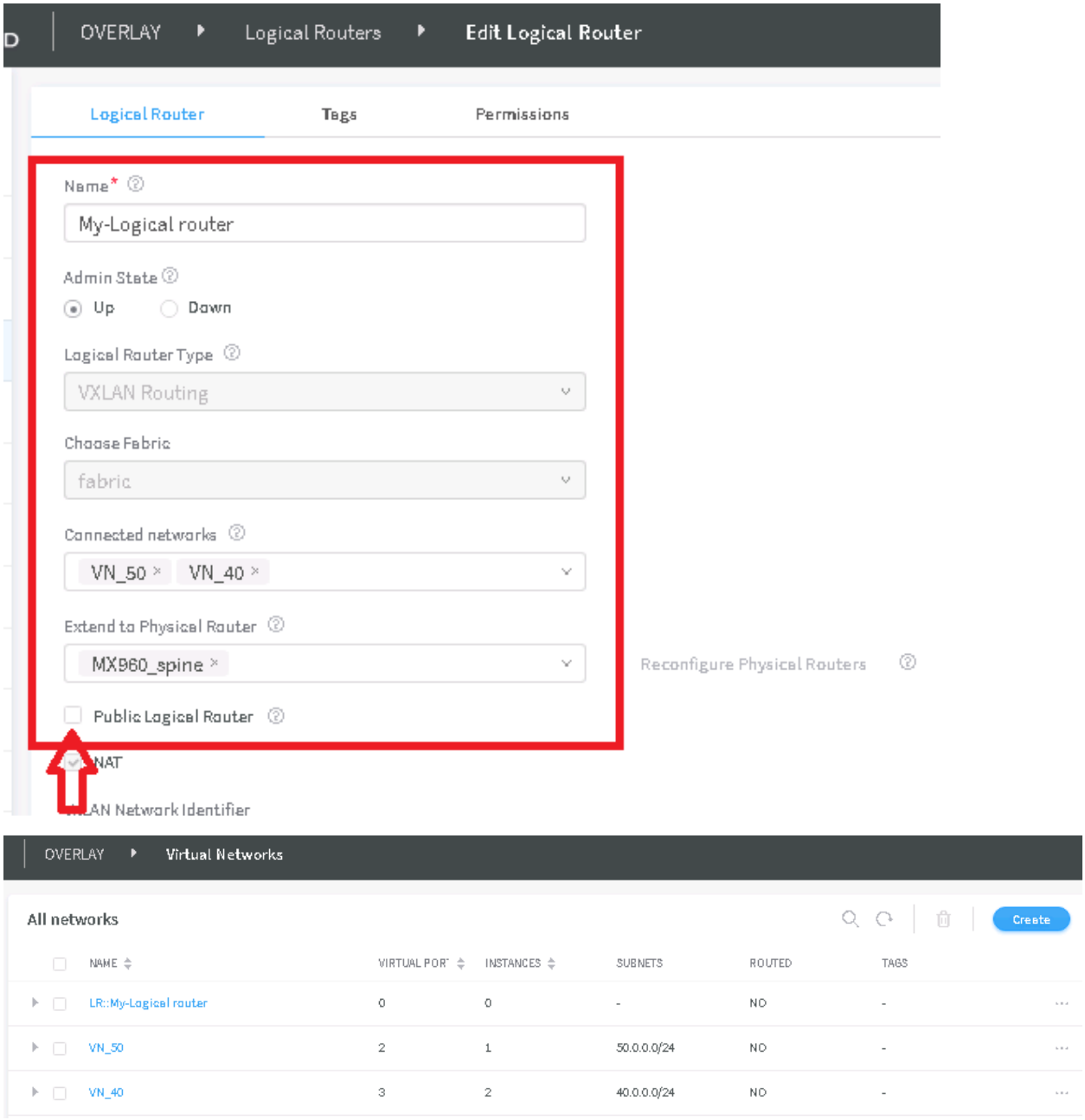
4.3. Create fabric & Create VN & Logical router

=> If you lookup the "Manual creating Fabric & enabling CRB-Gateway ", you have to enable "vxlan routing". This option was in "Admin" page. But in contrailv2005, it moved to 'logical-router' configuring page.




STEP 1
Create Fabric
STEP 2
Device discovery
Device credentials
Username*
root
Password*
contrail123
+ Add
Management subnets
CIDR*
172.27.122.195/32
Gateway
Enter valid IPv4
+ Add
Loopback subnets
CIDR*
3.3.3.71/32
+ Add


Assign role to 1 devices
Physical Role*
spine
Routing Bridging Roles
CRB-Gateway
Route-Reflector
Cancel
Assign



4.4. Be cautious

1). While setting 'Logical Router'

checkbox option "Public Logical Router". For this task, PLEASE DO NOT select the box. Just be empty. That option means to use Fabric as to "DC-gateway" using EVPN type, So in Fabric DC-gateway role has to be added.
If you see juniper web portal, officially it says "to use DC-gateway, please select that". But in my test it didn't work. For DC-gateway there is another method, we will cover that later Lab test.

2). MX/QFX version & product lists which support contrail

> Please check if your Spine and Leaf device has right version installed.

Contrail Networking Supported Hardware Platforms and Associated Roles And Node Profiles - TechLibrary - Juniper Networks

Table 2: Supported MX Series Routers

MX Device	Supported from Contrail Networking Release Supported from Junos OS Releases								
	Physical Roles		Overlay Roles				Gateway Roles		Special Role
	Leaf	Spine	ERB-UCAST-Gateway	CRB-Gateway	CRB-MCAST-Gateway	null	DC-Gateway	DCI-Gateway	Route-Reflector
MX80	5.0.2 17.3R3	5.0.2 17.3R3				5.1 18.1R3	5.0.2 17.3R3		
MX240, MX480, MX960	5.0.2 17.3R3	5.0.2 17.3R3	2003 18.4R2-S3	2003 18.4R2-S3	2003 18.4R2-S3	5.1 18.1R3	2003 (without SNAT) and 2005 (with SNAT) 18.4R2-S3	2005 18.4R2-S3	5.0.2 17.3R3

Table 1: Supported QFX Series Switches

QFX Device	Supported from Contrail Networking Release Supported from Junos OS Releases									
	Physical Roles			Overlay Roles					Gateway Roles	
	Leaf	Spine	Superspine	CRB-Access	CRB-GW	CRB-MCast-GW	ERB-UCast-GW	lean	DC Gateway	DCI Gatew
QFX5100-XX models	5.0.2 17.3R3	5.0.2 17.3R3		5.0.2 17.3R3				5.0.2 17.3R3		
QFX5110-48S-4C QFX5110-32Q	5.0.2 17.3R3	5.0.2 17.3R3		5.0.2 17.3R3	5.0.2 18.1R3		5.1 18.1R3	5.0.2 17.3R3		

4.4. If ping is not reachable between VM1(40 net) and VM3(50 net)

1). After you done setting above, if ping is unreachable, you have to check underlay environment. Maybe ping from compute node to MX/QFX loopback would be unreachable. One thing Important is when you set up contrail Fabric, it does for only overlay part. So for the underlay network, you have to set it properly so that among compute node and MX/QFX loopback must be reachable.

> On Compute2 node, do ping to MX960 loopback. If ping is unreachable, please adds Route properly.

```
[root@compute2-225 ~]# ping 3.3.3.71
[root@compute2-225 ~]# netstat -rn
 Kernel IP routing table
 Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
 0.0.0.0         172.27.122.254  0.0.0.0         UG        0 0          0 enp5s0f0
 100.0.0.0       0.0.0.0         255.255.255.0   U         0 0          0 vhost0
 169.254.0.3     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
 172.17.0.0      0.0.0.0         255.255.0.0     U         0 0          0 docker0
 172.27.122.0    0.0.0.0         255.255.255.0   U         0 0          0 enp5s0f0

[root@compute2-225 ~]# ip route add 3.3.3.0/24 via 100.0.0.240   <-=== 100.0.0.240 is EX switch Vlan interface ip (gateway = irb ip address)

[root@compute2-225 ~]# netstat -rn
 Kernel IP routing table
 Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
 0.0.0.0         172.27.122.254  0.0.0.0         UG        0 0          0 enp5s0f0
 3.3.3.0         100.0.0.240     255.255.255.0   UG        0 0          0 vhost0
 100.0.0.0       0.0.0.0         255.255.255.0   U         0 0          0 vhost0
 169.254.0.3     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
 172.17.0.0      0.0.0.0         255.255.0.0     U         0 0          0 docker0
 172.27.122.0    0.0.0.0         255.255.255.0   U         0 0          0 enp5s0f0

[root@compute2-225 ~]# ping 3.3.3.71
 PING 3.3.3.71 (3.3.3.71) 56(84) bytes of data.
 64 bytes from 3.3.3.71: icmp_seq=1 ttl=62 time=5.01 ms
 64 bytes from 3.3.3.71: icmp_seq=2 ttl=62 time=0.600 ms
```

2). If ping still not reachable, check if gre tunnel interface created properly.

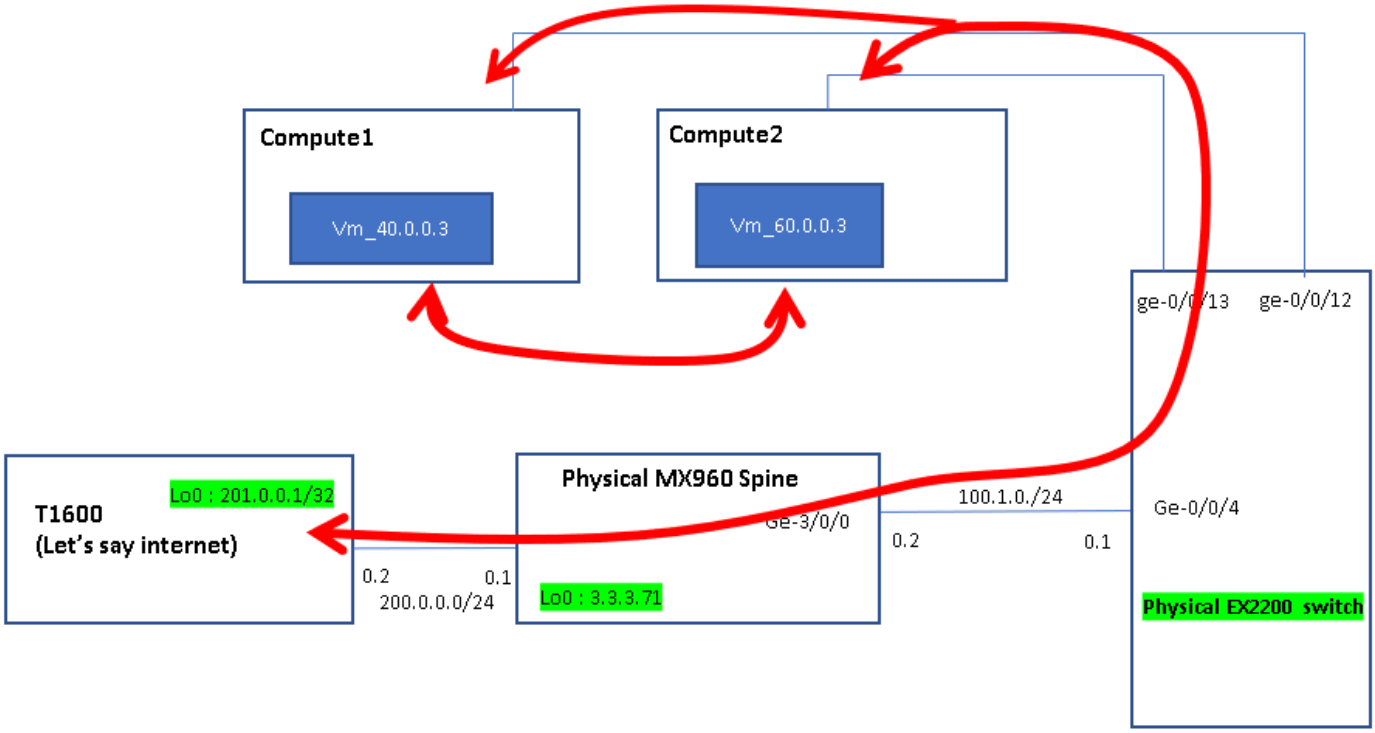
For L2 network, vxlan tunnel used.
But L3 network (using 'Extending Physical Router'), GRE tunnel used. So gre interface has to be up and check it's status installed correctly. ("show dynamic-tunnels database")

> How to enable GRE interface.

```
test@MX960_spine# show chassis
fpc 3 {
    pic 0 {
        tunnel-services {    <-=== please set tunnel service based on your FPC type properly.
            bandwidth 1g/10g;
        }
    }
}
```

5. Lab test #4 : Ping to Internet and Ping between VMs (L3 Gateway)

> Lab Topo



5.1. Key points at this lab.

1. In Fabric spine, you should add roles “DC-Gateway” and “CRB-Gateway”.
   So remove Fabric set in Lab#3, add new roles “DC-Gateway” and “CRB-Gateway” to spine.

2. Logical-Router setting view in GUI, you have to select “Public logical Router” option box. This enables you to inet.0 FBF. So that you can do internet. Please refer to Youtub below reference.

3. Key important, for internet, EVPN type5 used, so under L3 VPN VRF instance, 0.0.0.0/0 next-table inet.0 imported, and return traffic from internet , Filter Based Forwarding applied and leaked to VRF instance.
   Below picture shows importance concept, please reminded.

4. There is another method to send traffic to internet. If you studied JNCIE-DC, There it’s addressed. For L2 switching instance virtual-switch used, for L3 routing, instance virtual-router used.

5. MX and QFX show different behavior regarding tunnel interface.

   MX/vMX creates dynamic-tunnel interface for Spine, but vQFX doesn’t create instead VXLAN tunnel.

   1) For MX or vMX : below config pushed. you have to set/enable tunnel interface.

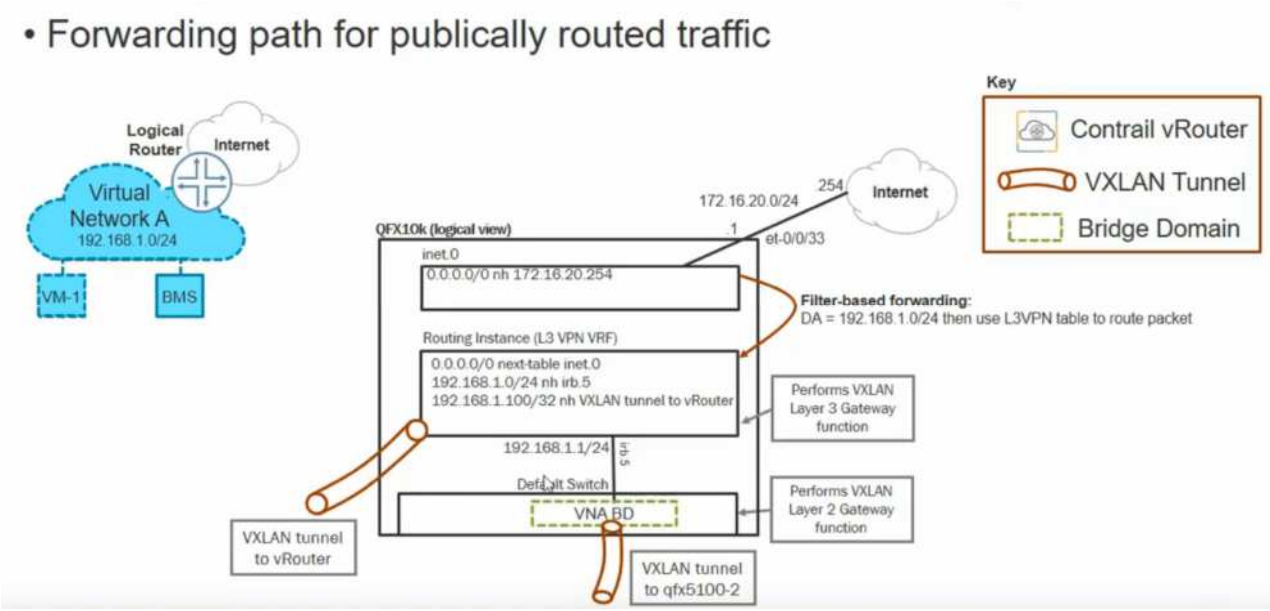
```
lab@vmx_re# show | display set | match dyna
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel source-address 1.1.1.1
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel udp
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel destination-networks 192.168.200.0/24

test@MX960_spine# show chassis
fpc 3 {
    pic 0 {
        tunnel-services {
            bandwidth 1g;
        }
    }
}
```

   2) For vQFX
      Dynamic-tunnel not created, but VXLAN tunnel interface created via physical interface
      Please refer to “Lab test#5” below

> Juniper Learning Byte. (How to set it up.)
https://youtu.be/MgXbC1OoI7c

5.2. Key functionality for DC-gateway

> Capture “Kye informatoin” from Juniper learning bye YouTube. (refer to above link)



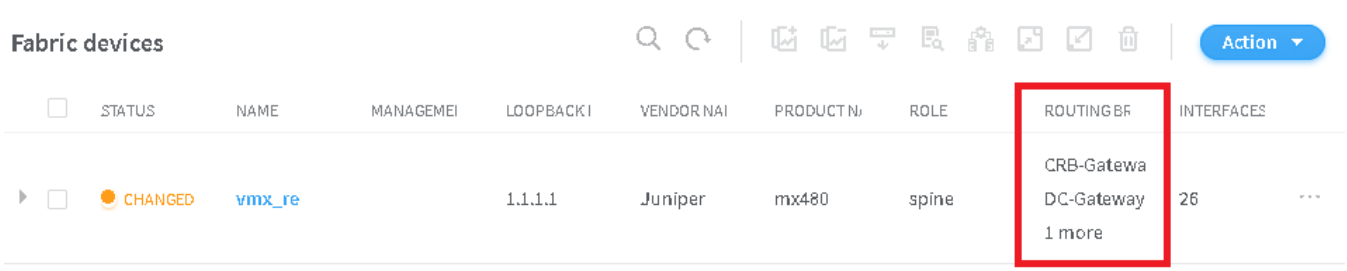
- There must be existing…
  - Contrail cluster with Contrail Command user interface in place
  - VXLAN routing enabled
  - Onboarded IP fabric with EVPN/VXLAN overlay signaling established
    - Spine nodes assigned CRB-Gateway role and DC-Gateway role
    - Leaf nodes assigned CRB-Access role

5.3. Setting up Fabric configuration & correction

1). Remove ‘Fabric’ config of Contrail-command.

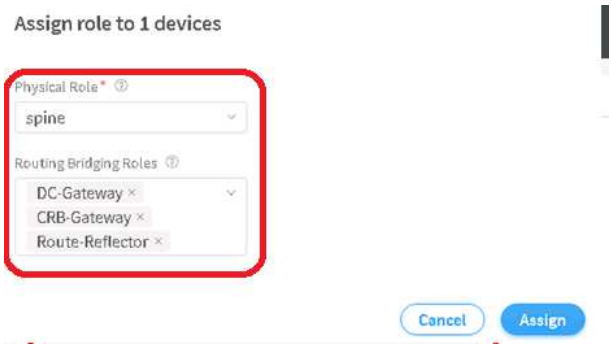
While you are Fabric setting up, please run Fabric ansible monitoring for debugging purpose.
[root@controller-222 ~]# tail -f /var/log/contrail/contrail-fabric-ansible-playbooks.log

2). For exist VM and VN, please keep them. You need to remove only ‘Logical-Router’ and ‘Fabric’ configuration.

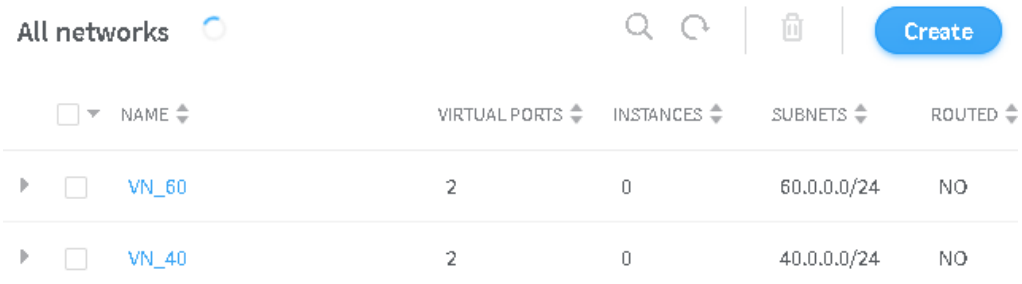


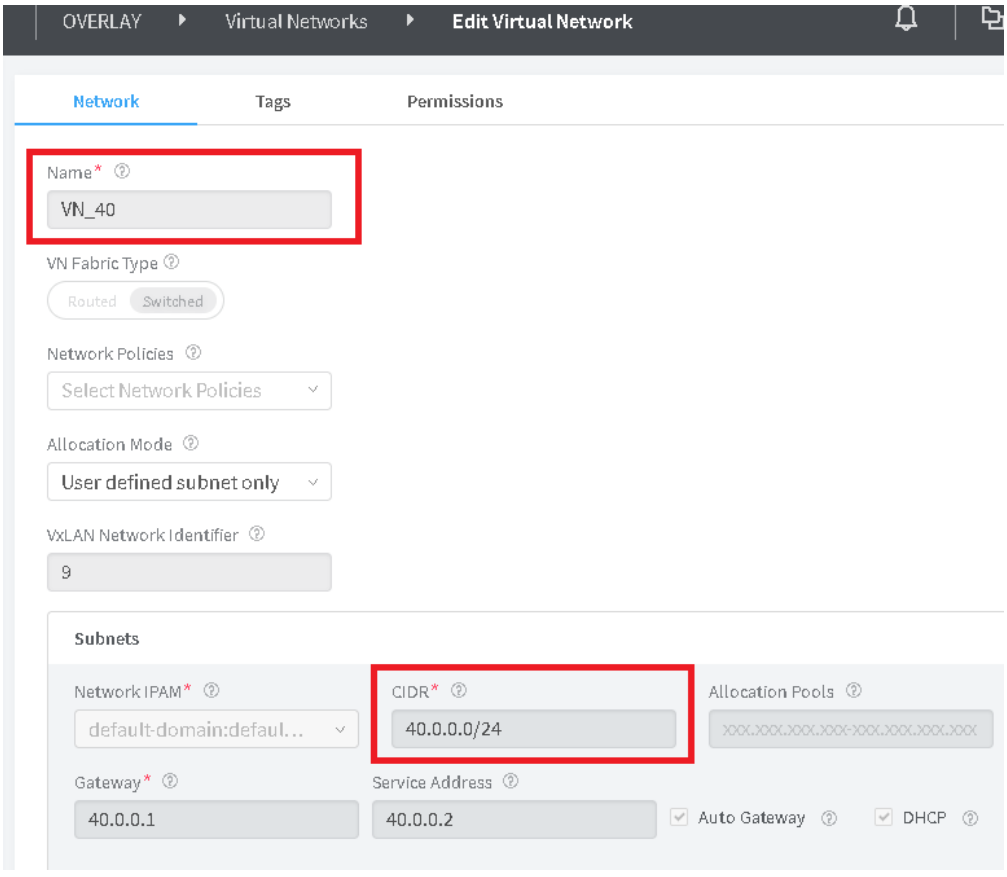
**** Please be aware of that.**

CIDR for Management and Loopback, please subnet '/32'. If you put /24, Fabric tries to discover all equip under /24. Sometimes, it causes Fabric registration failure. So please be specific.

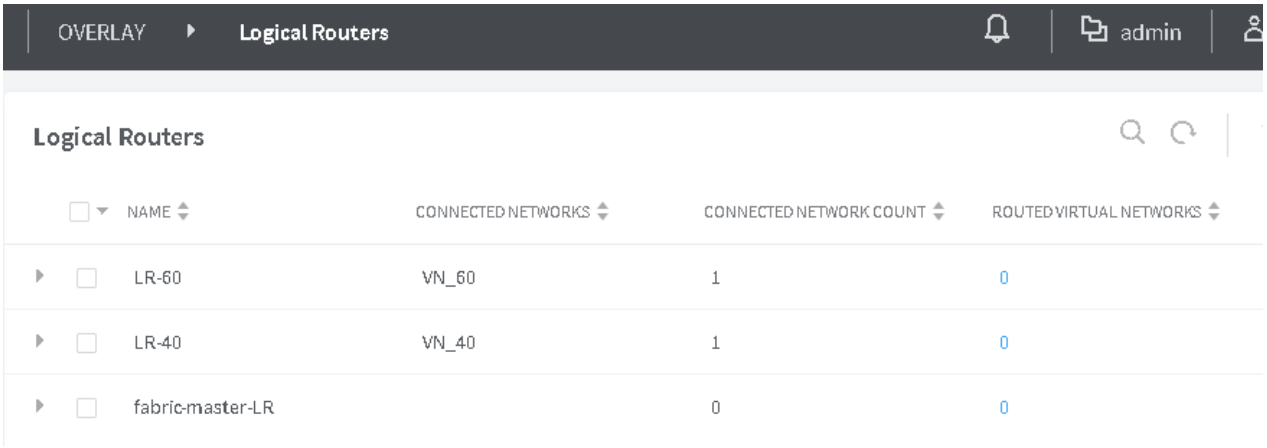


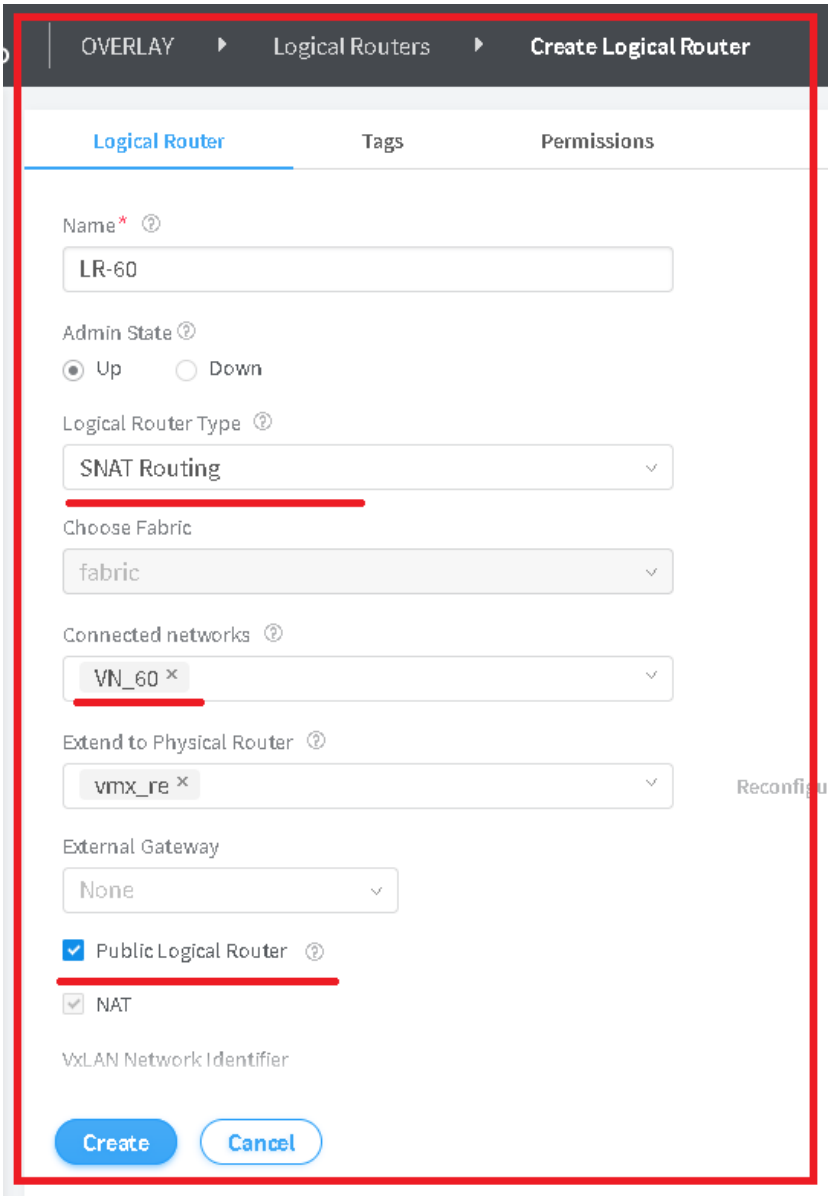
> Create just normal 2 Virtual Networks



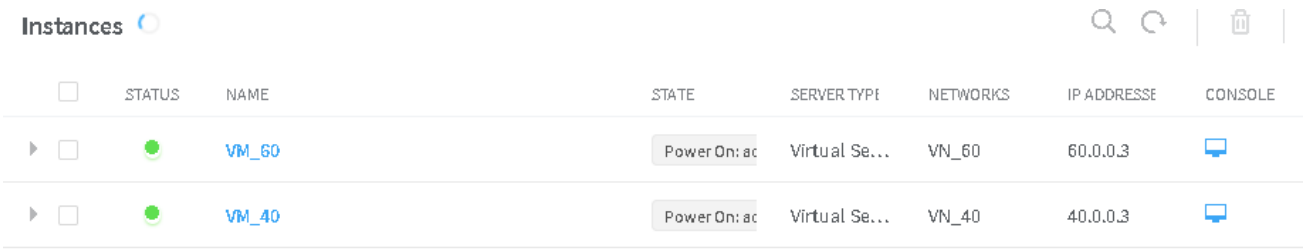


> Create 2 Logical-Router, and assign each VN to each Logical-Router





> Create 2 instances.



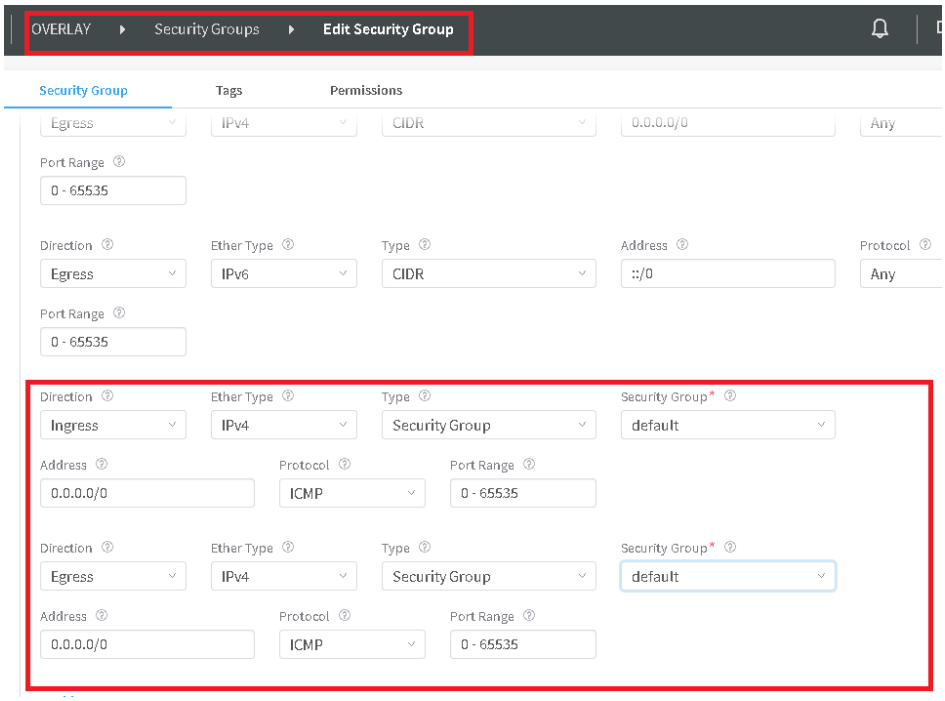
5.4. After all setting done, do ping test between VMs and to Internet

1). Once you done, ping test VMs(60.0.0/24, 40.0.0/24) to Internet(201.0.0.1 lo of T1600), it must be reachable.

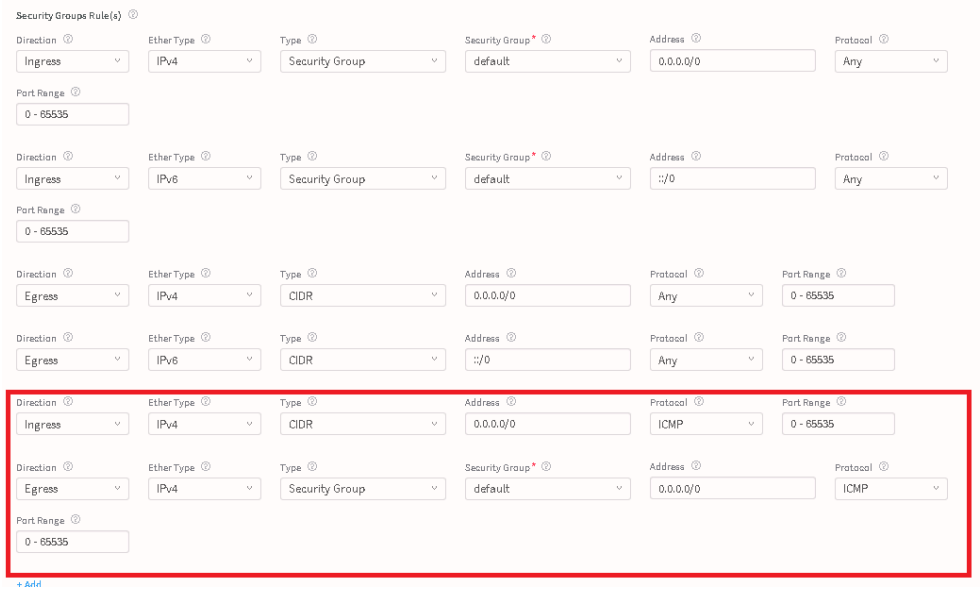
2). If at Lab test#1, if you set 'Security group' like below, ping between VMs (60.0.0.3 ←-->40.0.0.3) is unreachable.

Please refer to below correct 'Security group' set.

> bad 'Security group' set



> Correct 'Security group'



3). After correcting 'Security group' set, if ping still not reachable, please check Underlay between compute node and controller node.

Ping between controller/compute and lo0 of MX, must be reachable.

Contrail only cares for overlay. So underlay must be set implemented by operator manually.

> on Compute2, ping is not reachable to MX960 loopback. After adding route, ping successful

```
[root@test:test ~]# ping 1.1.1.1

[root@test:test ~]# yum -y install net-tools

[root@test:test ~]# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.49.127.254   0.0.0.0         UG        0 0          0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 eth1
169.254.0.1     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
169.254.0.3     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
172.17.0.0      0.0.0.0         255.255.0.0     U         0 0          0 docker0
192.168.200.0   0.0.0.0         255.255.255.0   U         0 0          0 vhost0
192.168.255.0   0.0.0.0         255.255.255.0   U         0 0          0 eth1

[root@test:test ~]# ip route add 1.1.1.1/32 via 192.168.200.254   <=== 192.168.200.254   is gateway ip addr

[root@test:test ~]# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.49.127.254   0.0.0.0         UG        0 0          0 eth0
1.1.1.1         192.168.200.254 255.255.255.255 UGH       0 0          0 vhost0
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 eth1
169.254.0.1     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
169.254.0.3     0.0.0.0         255.255.255.255 UH        0 0          0 vhost0
172.17.0.0      0.0.0.0         255.255.0.0     U         0 0          0 docker0
```

```
192.168.200.0     0.0.0.0         255.255.255.0   U         0 0          0 vhost0
192.168.255.0     0.0.0.0         255.255.255.0   U         0 0          0 eth1


[root@test:test ~]# ping 1.1.1.1
 PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
 64 bytes from 1.1.1.1: icmp_seq=1 ttl=62 time=339 ms
 64 bytes from 1.1.1.1: icmp_seq=2 ttl=62 time=124 ms
```

4). But if still, ping not reachable, check if gre tunnel interface created properly. It's only applicable to MX/vMX. For QFX you don't need gre tunnel interface. VXLAN tunnel interface used on physical interface.

At this lab, for L2 switching vxlan encapsulation used, but for L3 routing between vRouter and MX/vMX, gre tunnel interface used

> for MX/vMX, please check if gre interface is up or not.

```
test@MX960_spine# show chassis
  fpc 3 {
        pic 0 {
              tunnel-services {
                    bandwidth 1g;
              }
        }
    }

test@MX960_spine#run show dynamic-tunnels database
```

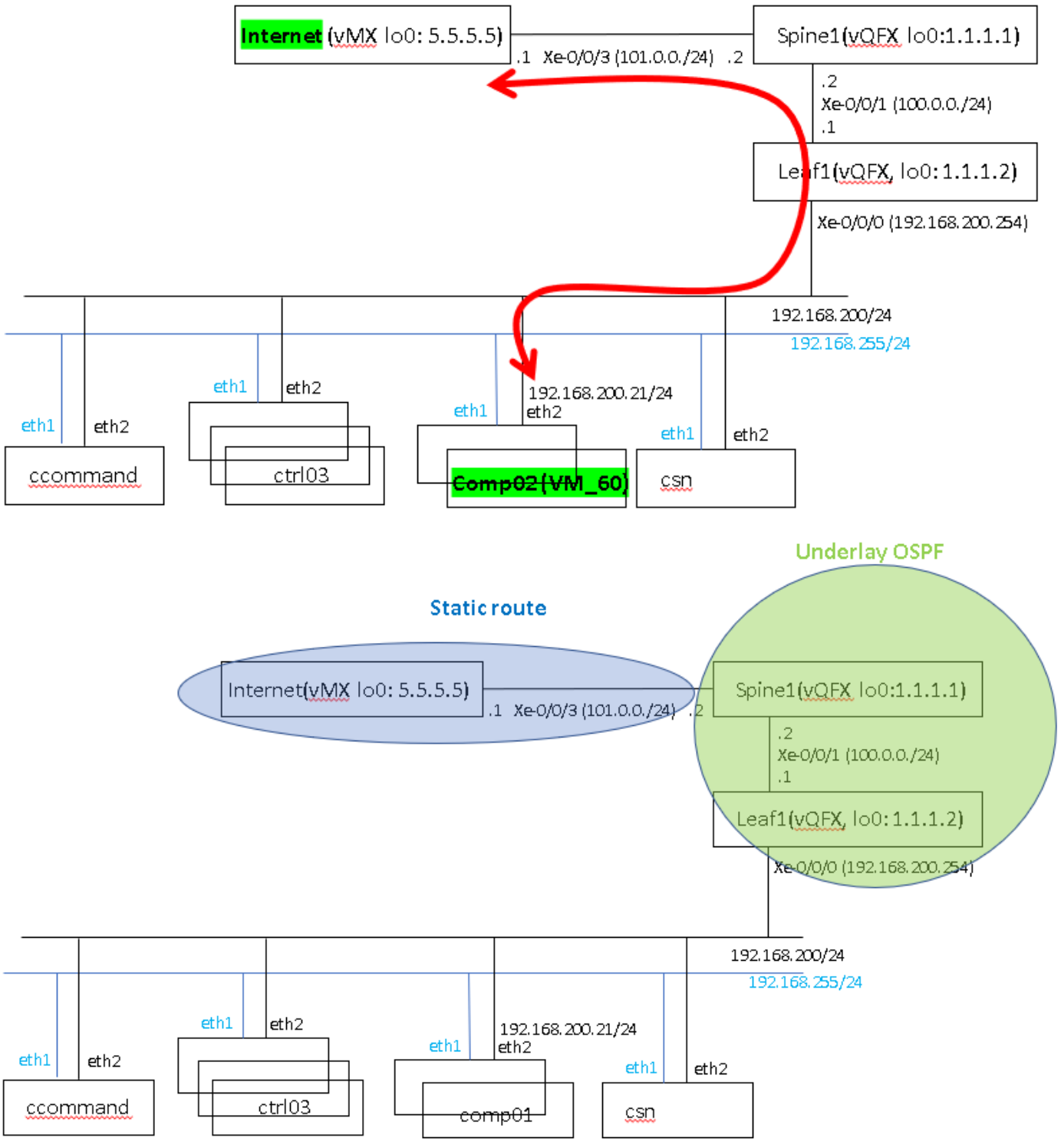
5). But ping still not reachable, for troubleshooting in detail, I will release next Doc 'Finding vRouter NH', and refer to that

6. Lab test #5 : Ping to Internet (Spine & leaf)

** This is new lab environment, so different from previous Lab test#1,#2, #3,#4.

Basically not that much different on this test, but only Spine/Leaf used

> Lab Topo



6.1. Important & be minded for MX vs QFX spine role.

When you use MX as Spine for CRB-Gateway or DC-Gateway Role, GRE tunnel interface creation needed, so relevant configuration (“ set chassis fpc X pic X tunnel-services”) required.
But in CASE QFX, GRE tunnel is not used, but VXLAN tunnel created via physical interface.

* For MX or vMX, below config being pused by contrail-controller

```
lab@vmx_re# show | display set | match dyna
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel source-address 1.1.1.1
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel udp
set groups __contrail_overlay_bgp__ routing-options dynamic-tunnels _contrail_udp_tunnel destination-networks 192.168.200.0/24
```

* For QFX
GRE tunnel interface not created, but VXLAN tunnel interface created via physical interface

6.2. Configurations for an infrastructure. (vMX, vQFX, Computes, Controllers)

> vMX

```
set interfaces ge-0/0/3 unit 0 family inet address 101.0.0.1/24
set interfaces lo0 unit 0 family inet address 5.5.5.5/32
set routing-options static route 40.0.0.0/24 next-hop 101.0.0.2
set routing-options static route 60.0.0.0/24 next-hop 101.0.0.2
```

> vQFX : Spine1

```
set interfaces xe-0/0/1 unit 0 family inet address 100.0.0.2/24
set interfaces xe-0/0/3 unit 0 family inet address 101.0.0.2/24
set interfaces lo0 unit 0 family inet address 1.1.1.1/32
set routing-options static route 1.1.1.3/32 next-hop 101.0.0.1
set routing-options static route 5.5.5.0/24 next-hop 101.0.0.1
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ospf area 0.0.0.0 interface xe-0/0/1.0
```

> vQFX : leaf1

```
set interfaces xe-0/0/0 unit 0 family inet address 192.168.200.254/24
set interfaces xe-0/0/1 unit 0 family inet address 100.0.0.1/24
set interfaces lo0 unit 0 family inet address 1.1.1.2/32
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ospf area 0.0.0.0 interface xe-0/0/1.0
set protocols ospf area 0.0.0.0 interface xe-0/0/0.0
```

> Compute node(comp01/02), Controller node (ctrl01/02/03)

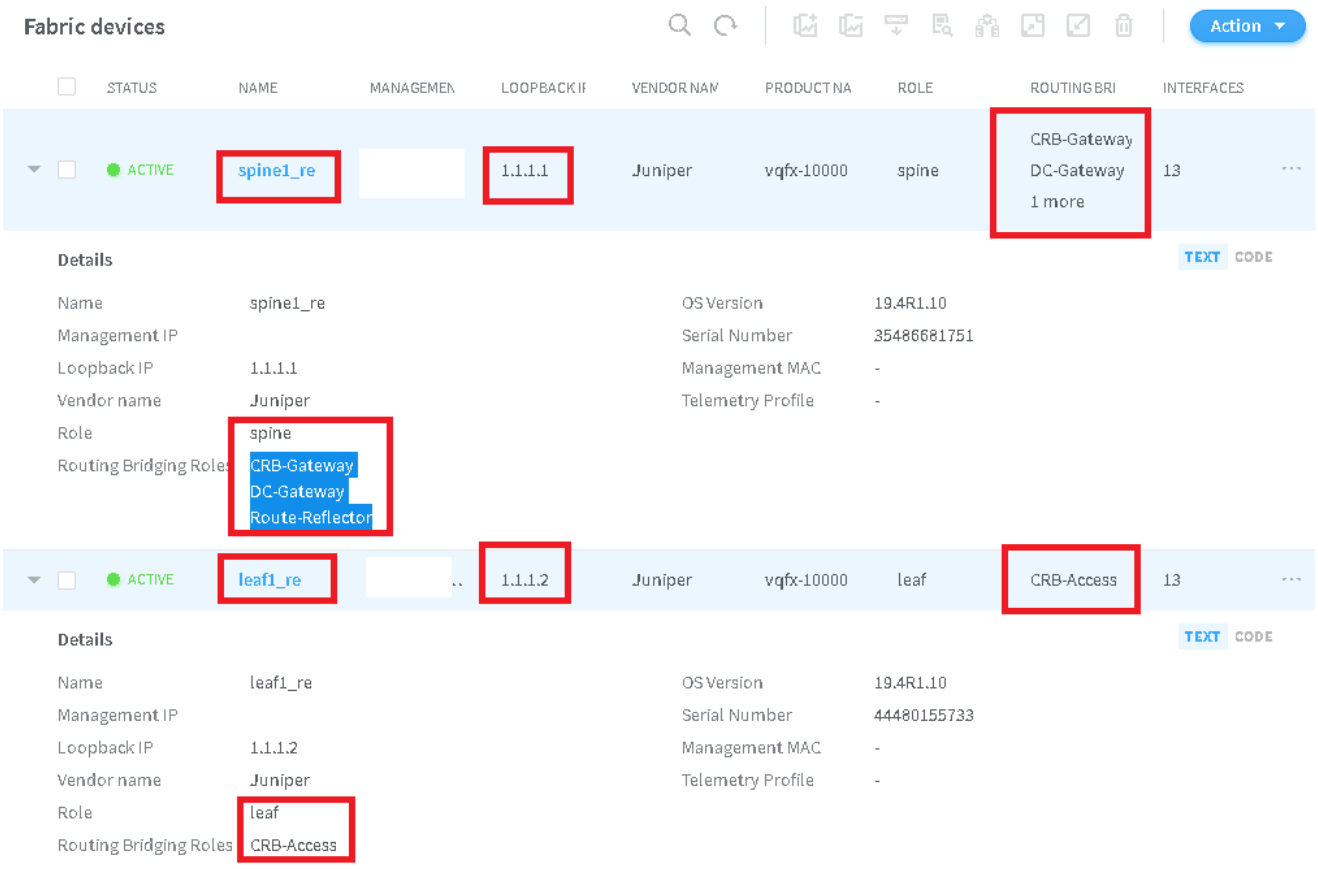
```
[root@test:test ~]#ip route add 1.1.1.0/24 via 192.168.200.254
[root@test:test ~]# netstat -rn
 Kernel IP routing table
 Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
  ~
1.1.1.0          192.168.200.254 255.255.255.0   UG        0 0          0 vhost0
  ~
```

[root@test:test ~]# ping 1.1.1.1 **<== ping must be reachable to Spine/leaf lo0 address from compute and controller.**
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=62 time=119 ms

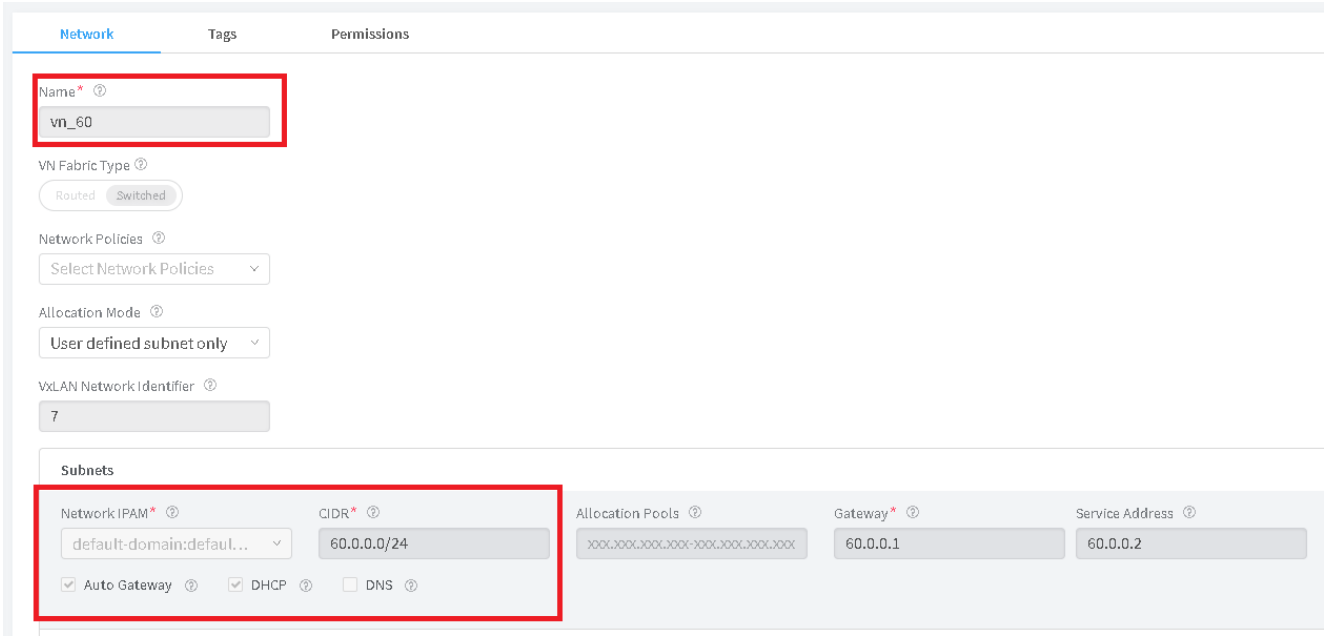
6.3. Setting up Fabric configuration & correction

1) Add Fabric Spine and leaf, and assign each role properly.

**** Then check IBGP connection properly established with Spine/Leaf and Controllers.**

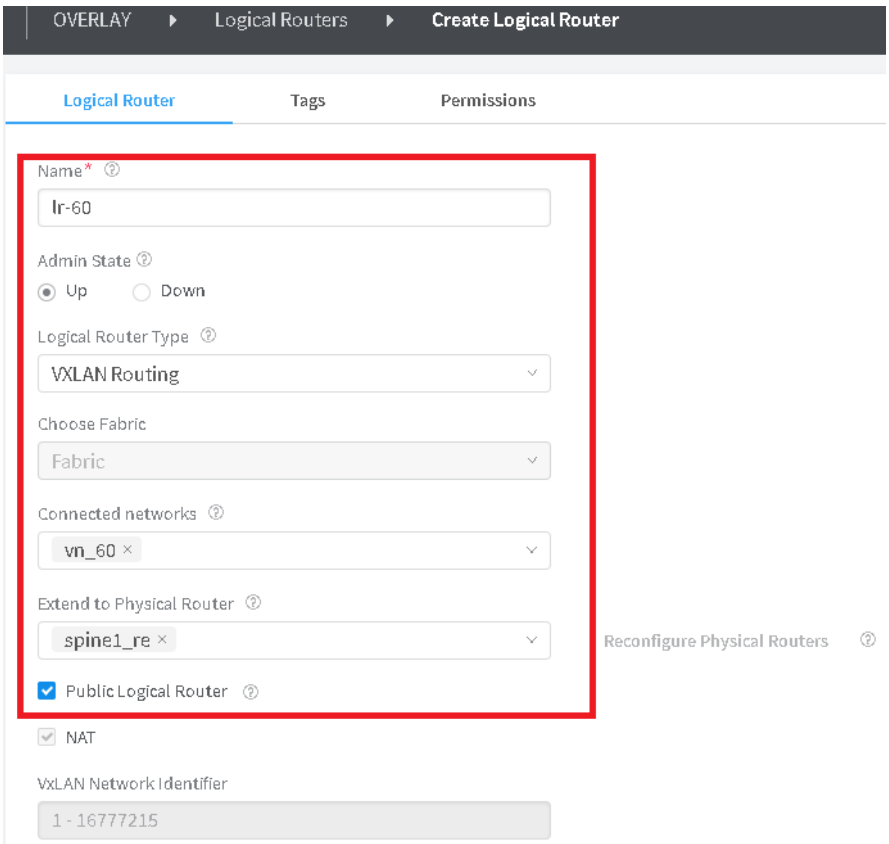


2). Create VN(Virtual Network)

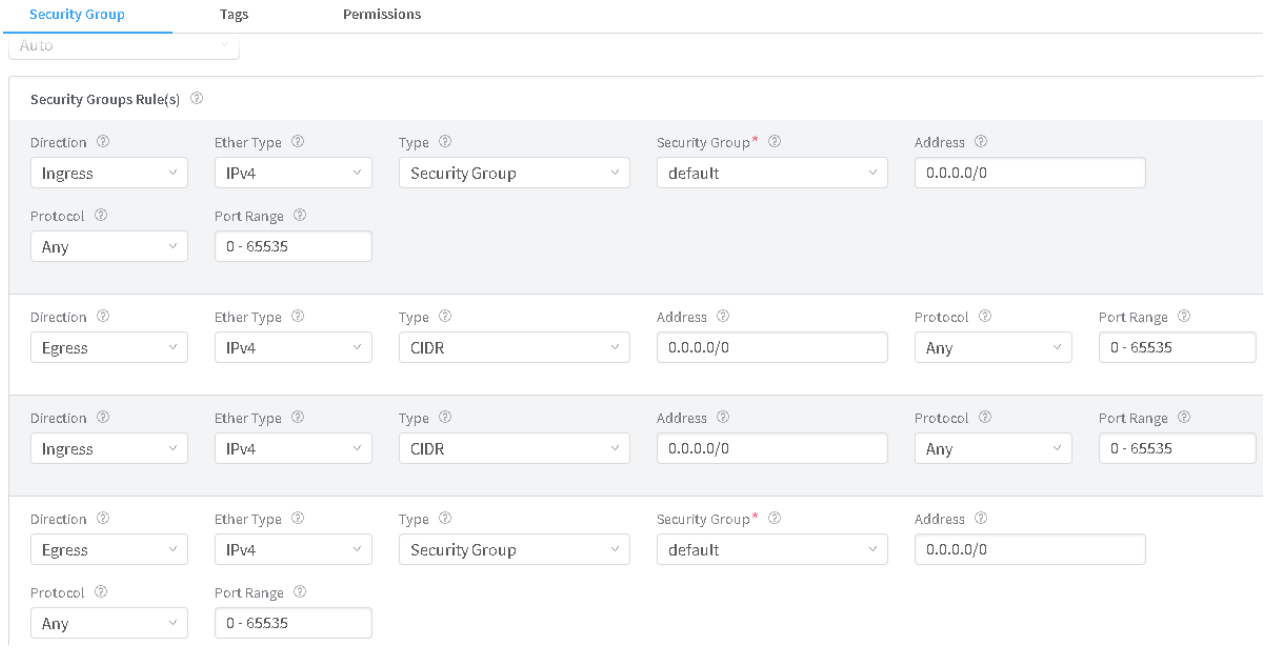


3) Create logical-router

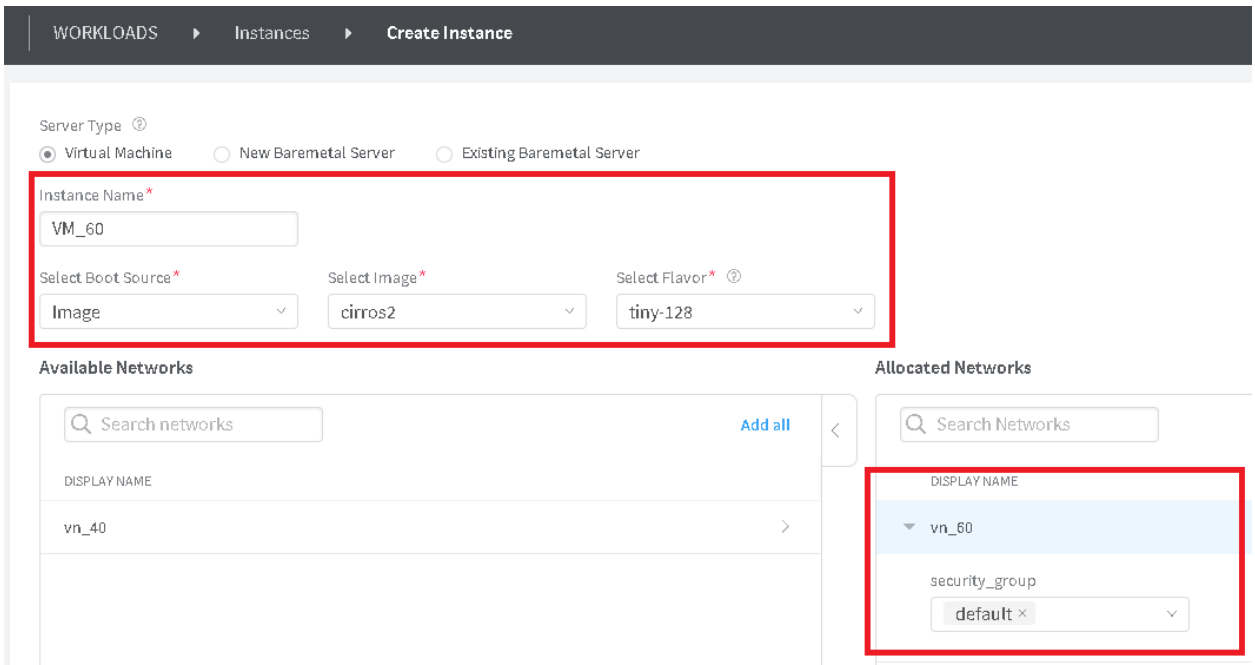
=> please be aware of “Public Logical Router” option below. That is for evpn-type5. So we can send traffic to internet by importing default route.



4). Check “Security Group”, allow Ingress/Egress



5) create Instance “VM_60”



6.3. What to check

1). Check Spine and Leaf node made IBGP connections with controllers.

```
lab@spine1_re# run show bgp summary
Threading mode: BGP I/O
Groups: 1 Peers: 4 Down peers: 0
Table            Tot Paths  Act Paths Suppressed    History Damp State    Pending
bgp.rtarget.0
                        29         17          0          0          0          0
bgp.l3vpn.0
                         8          6          0          0          0          0
bgp.l3vpn-inet6.0
                         0          0          0          0          0          0
bgp.evpn.0
                        32         16          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State|#Active/Received/Accepted/Damped...
1.1.1.2               64512        276        284       0       0        2:01:57 Establ   <----- with Leaf
  __default_evpn__.evpn.0: 0/0/0/0
  bgp.evpn.0: 0/0/0/0
  bgp.rtarget.0: 0/1/1/0
  default-switch.evpn.0: 0/0/0/0
192.168.200.11        64512        239        263       0       0        1:47:48 Establ   <--- Controller1
  __contrail_LR-40_5dae0b2b-337f-4733-bbdc-21e66a5deae5.evpn.0: 1/1/1/0
  __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.evpn.0: 1/1/1/0
  __default_evpn__.evpn.0: 0/0/0/0
  bgp.evpn.0: 16/16/16/0
  bgp.l3vpn-inet6.0: 0/0/0/0
  bgp.l3vpn.0: 4/4/4/0
  bgp.rtarget.0: 7/12/12/0
  default-switch.evpn.0: 10/10/10/0
192.168.200.12        64512        239        276       0       0        1:47:32 Establ   <--- Controller2
  __contrail_LR-40_5dae0b2b-337f-4733-bbdc-21e66a5deae5.evpn.0: 0/1/1/0
  __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.evpn.0: 0/1/1/0
  __default_evpn__.evpn.0: 0/0/0/0
  bgp.evpn.0: 0/16/16/0
  bgp.l3vpn-inet6.0: 0/0/0/0
  bgp.l3vpn.0: 2/4/4/0
  bgp.rtarget.0: 7/12/12/0
  default-switch.evpn.0: 0/10/10/0
192.168.200.13        64512        220        255       0       0        1:47:09 Establ   <--- Controller3
  __default_evpn__.evpn.0: 0/0/0/0
  bgp.evpn.0: 0/0/0/0
  bgp.l3vpn-inet6.0: 0/0/0/0
  bgp.l3vpn.0: 0/0/0/0
  bgp.rtarget.0: 3/4/4/0
  default-switch.evpn.0: 0/0/0/0
```

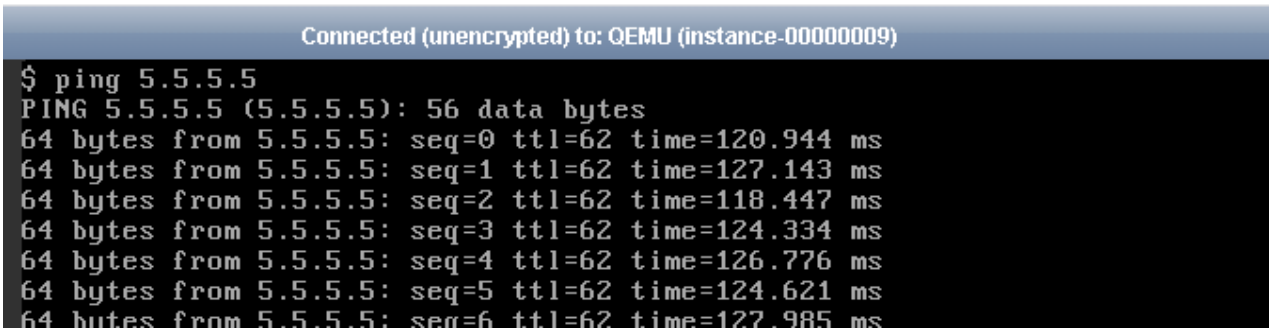
> Check routing-instance , table and VXLAN tunnel created properly.

```
lab@spine1_re# run show route table ?
Possible completions:
  <table>              Name of routing table
  :vxlan.inet.0
  __contrail_LR-40_5dae0b2b-337f-4733-bbdc-21e66a5deae5.evpn.0
  __contrail_LR-40_5dae0b2b-337f-4733-bbdc-21e66a5deae5.inet.0
  __contrail_LR-40_5dae0b2b-337f-4733-bbdc-21e66a5deae5.inet6.0
  __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.evpn.0   <---- EVPN type5
  __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet.0   <-- vrf routing-instance for VM_60
  __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet6.0
```

```
   __default_evpn__.evpn.0
   bgp.evpn.0
   bgp.l3vpn.0
   bgp.rtarget.0
   default-switch.evpn-mcsn.1
   default-switch.evpn.0
   inet.0
   inet6.0
{master:0}[edit]
lab@spine1_re# run show route table __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet.0

__contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
@ = Routing Use Only, # = Forwarding Use Only
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0               *[Static/5] 00:25:28
                                to table inet.0
60.0.0.0/24             *[Direct/0] 00:25:28
                          >     via irb.7
60.0.0.1/32             *[Local/0] 00:25:28
                                Local via irb.7
60.0.0.3/32             *[EVPN/170] 00:25:28
                          >     to 100.0.0.1 via xe-0/0/1.0      ←=== xe-0/0/1 is Vxlan tunnel between Spine and Compute node vRouter
60.0.0.5/32             *[Local/0] 00:25:28
                                Local via irb.7
172.16.0.1/32           *[Static/5] 00:25:28
                                Discard


{master:0}[edit]
lab@spine1_re# run show route 60.0.0.3 table __contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet.0

__contrail_lr-60_85a1c66c-dae4-4675-880b-c5d7d3f30cfa.inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
60.0.0.3/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 60.0.0.3/32 -> {composite(1817)}
            *EVPN      Preference: 170/-201
                       Next hop type: Indirect, Next hop index: 0
                       Address: 0xc65fef0
                       Next-hop reference count: 2
                       Next hop type: Router, Next hop index: 1731
                       Next hop: 100.0.0.1 via xe-0/0/1.0, selected
                       Session Id: 0x0
                       Protocol next hop: 192.168.200.21
                       Composite next hop: 0xc3d81c0 1817 INH Session ID: 0x0
                          VXLAN tunnel rewrite:
                             MTU: 0, Flags: 0x0
                             Encap table ID: 0, Decap table ID: 10
                             Encap VNI: 12, Decap VNI: 12
                             Source VTEP: 1.1.1.1, Destination VTEP: 192.168.200.21
                             SMAC: 02:05:86:71:24:00, DMAC: 56:68:a3:16:17:e0
                       Indirect next hop: 0xcb48904 131079 INH Session ID: 0x0
                       State: <Active Int Ext>
                       Age: 26:02          Metric2: 2
                       Validation State: unverified
                       Task: __contrail_lr-60_85a1c66c-dae4-4675-880b
                       Announcement bits (1): 3-KRT
                       AS path: ?
                       Communities: target:64512:8000014 encapsulation:vxlan(0x8) mac-mobility:0x0 (sequence 1) router-
   mac:56:68:a3:16:17:e0 evpn-etree:0x0:root (label 0) unknown type 0x8004:0xfc00:0x7a1202 unknown type 0x8071:0xfc00:0xc
                       Composite next hops: 1
                                Protocol next hop: 192.168.200.21 Metric: 2
```

```
Composite next hop: 0xc3d81c0 1817 INH Session ID: 0x0
  VXLAN tunnel rewrite:                    <====== vxlan tunnel created between Spine and compute node
    MTU: 0, Flags: 0x0
    Encap table ID: 0, Decap table ID: 10
    Encap VNI: 12, Decap VNI: 12
    Source VTEP: 1.1.1.1, Destination VTEP: 192.168.200.21    <==== compute node : 192.168.200.21
    SMAC: 02:05:86:71:24:00, DMAC: 56:68:a3:16:17:e0
Indirect next hop: 0xcb48904 131079 INH Session ID: 0x0
Indirect path forwarding next hops: 1
        Next hop type: Router
        Next hop: 100.0.0.1 via xe-0/0/1.0
        Session Id: 0x0
        192.168.200.0/24 Originating RIB: inet.0
          Metric: 2         Node path count: 1
          Forwarding nexthops: 1
                Nexthop: 100.0.0.1 via xe-0/0/1.0
                Session Id: 0
```

6.4. After all setting done, do ping test between VM_60 and Internet



7. Contrail Networking Overview <Figure>

Figure 1: Contrail Networking Overview