CIAB Internetworking for Multi-Tenant, Multi-Node, Multi-Cloud/Hybrid Environment using Node based BGP VRF to advertise Nodal based LXD Tenant to all Tenant VPN Members

by Brian Mullan (bmullan.mail@gmail.com) 1/20/2020

Installation

Overview Steps

1. Create each Ubuntu 18.04 LTS based Node used to Host Tenant LXD containers. This can be Cloud or Hybrid Cloud ased servers.

2. Install SNAP LXD

3. Install Nebula Network Overlay application on 1 Nebula Lighthouse Node and X number of Nebula Network Overlay Nodes.

4. Install Quagga on every Nebula Node including the Nebula Lighthouse

5. Create custom LXD Bridge & Profile for each Tenant.

6. Create X number of Tenant containers using that Tenant's custom LXD Bridge and Profile.

7. Configure Quagga on each Nebula Node to advertise the each Tenant's LXD subnet and Tenant LXD traffic to any other Nebula Node that serves that Tenant.

   NOTE: *The Nebula Lighthouse Node will not have to advertise any LXD networks because it will not be running LXD tenant containers*

   1. Each Tenant will be assigned a different Autonomous System (AS) Number in BGP across the entire Nebula Network Overlay of Nodes
   2. The BGP config will use the Nebula TUN Interface (Nebula1) IP address on each Node as the "Network" IP.

Step 1

Install Quagga & BGP on each Nebula Node & Lighthouse

> ***NOTE: The following is a Quagga/BGP Install Bash Script. On each Node (including the Nebula Lighthouse) create a file (re like "mk-bgp.sh") and copy the following into that file, save it and make it executable (re chmod +x mk-bgp.sh). Then execute that bash script to install everything on each Node to enable Quagga and BGP Routing.***

```
#!/bin/bash

# Install the Quagga routing daemon:

sudo apt-get install quagga quagga-doc -y

sudo mkdir -p /var/log/quagga && sudo chown quagga:quagga /var/log/quagga

# Enable IPv4 and IPv6 Unicast Forwarding:

echo "net.ipv4.conf.all.forwarding=1" | sudo tee -a /etc/sysctl.conf
echo "net.ipv4.conf.default.forwarding=1" | sudo tee -a /etc/sysctl.conf

sed 's/#net.ipv6.conf.all.forwarding=1/net.ipv6.conf.all.forwarding=1/g' /etc/sysctl.conf | sudo tee \
/etc/sysctl.conf

echo "net.ipv6.conf.default.forwarding=1" | sudo tee -a /etc/sysctl.conf

sudo sysctl -p

# Enable IPv4 Multicast Forwarding:

echo "net.ipv4.conf.all.mc_forwarding=1" | sudo tee -a /etc/sysctl.conf
echo "net.ipv4.conf.default.mc_forwarding=1" | sudo tee -a /etc/sysctl.conf

sudo sysctl -p

# Copy the template Configuration files that get installed
# when we installed quagga-doc above.

sudo cp /usr/share/doc/quagga-core/examples/zebra.conf.sample /etc/quagga/zebra.conf
sudo cp /usr/share/doc/quagga-core/examples/bgpd.conf.sample /etc/quagga/bgpd.conf
sudo cp /usr/share/doc/quagga-core/examples/vtysh.conf.sample /etc/quagga/vtysh.conf

# Set the Owner & Group to "quagga"
```

```
sudo chown quagga.quagga /etc/quagga/*.conf

# Change Permissions to 640 for all config files

sudo chmod 640 /etc/quagga/*.conf

# Prevent a daemon from running:

sudo unlink /etc/systemd/system/multi-user.target.wants/bgpd.service
sudo unlink /etc/systemd/system/multi-user.target.wants/zebra.service

#sudo unlink /etc/systemd/system/multi-user.target.wants/isisd.service
#sudo unlink /etc/systemd/system/multi-user.target.wants/ospf6d.service
#sudo unlink /etc/systemd/system/multi-user.target.wants/ospfd.service
#sudo unlink /etc/systemd/system/multi-user.target.wants/pimd.service
#sudo unlink /etc/systemd/system/multi-user.target.wants/ripd.service
#sudo unlink /etc/systemd/system/multi-user.target.wants/ripngd.service

# Reinstate a daemon to run:

sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/bgpd.service
sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/zebra.service

#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/isisd.service
#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/ospf6d.service
#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/ospfd.service
#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/pimd.service
#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/ripd.service
#sudo ln -st /etc/systemd/system/multi-user.target.wants /lib/systemd/system/ripngd.service

# Restart the daemons:

sudo systemctl restart zebra.service
sudo systemctl restart bgpd.service

#sudo systemctl restart pimd.service
#sudo systemctl restart ripd.service
#sudo systemctl restart ripngd.service
#sudo systemctl restart ospf6d.service
#sudo systemctl restart isisd.service
#sudo systemctl restart ospfd.service

exit 0
```

Step 2

Node 0 Quagga BGP Configuration

Nebula Lighthouse (Node0) will be the Master BGP Node.

What this means is that the ***Nebula Lighthouse Node (Node 0)*** will have the BGP Configuration that will also include the **Virtual Routing and Forwarding (VRF)** statements

NOTE: Edit the /etc/quagga/bgpd.conf file on Node0 to look like the following...

```
============================================================
# CIAB Ubuntu VxLAN OVS BGP VRF Configuration
# NODE1 configuration - /etc/quagga/bgpd.conf
# Master Node which has VRFs configured for all
# other LXD Server/Host Nodes
#
# (c) brian mullan (ciab)
#============================================================

#============================================================
# Quagga BGPd configuration file for CIAB Nebula Lighthouse (Node0).
#
# Node1 is the node with the Virtual Routing and Forwarding (re VRFs) defined
# for each of the other LXD Host/Server nodes in the rest of  the Network.
#============================================================
hostname node0
password ciab2020
enable password ciab2020

#============================================================
# bgp multiple-instances...
#============================================================
# Define Autonomous System (AS) number for this specific BGP
# network (re 64512)
#
# NOTE:
# there can be more than 1 if multi-tenant -or- multi-network
# configuration that Node0 can belong to.
#
# If more than one AS is defined then NODE0 BGP would know to
# participate in more than one AS Network.
#============================================================

#===={Define 2 different Tenant BGP AS numbers}===
```

router bgp 64512 # **Tenant1**
router bgp 64513 # **Tenant2**

#===
ID "this" BGP Node (Node0) by its Nebula TUN IP address (Nebula1 interface IP)
#===

bgp router-id 10.10.200.254

#===
As ***Node0 is a master node*** and using its VRFs to provide
Routing information to each of its sub-ordinate Peers.
#
Add BGP Peer nodes. One entry for each Nebula LXD Host/Server node
using that Host/Server's Nebula TUN interface IP address and the same
Autonomous System (AS) number as defined above.
#
NOTE:
Again, if this is a Multi-Tenant or a Multi-Network
configuration EACH of these "could" be in a different AS
Number which would effectively separate Traffic like a VPN
would do except this would also Route via BGP.
#
This Node0 would have to have more than 1
"router bgp <AS number>" statement though so it'd know to
participate in more than one BGP AS network routing operation.
#===
Format:
neighbor <Nebula Node1's Nebula1 TUN IP> <AS number>
neighbor <Nebula Node2's Nebula1 TUN IP> <AS number>
#
#==={example for a **Tenant1** and a **Tenant2**]===
 # The example assumes there is are 2 Tenants on each Node so each Node would
 # have 2 AS numbers, one for each Tenant's custom LXD bridge IP address on that
 # node.
#
NOTE: In configuration of BGP on Node 0 (the BGP Master Node w the VRFs)
"conceptually" <u>think of each BGP AS number</u> *as a Tenant ID*.

#==={for **Tenant1 = AS 64512**}===
#neighbor **10.10.200.1** remote-as 64512
#neighbor **10.10.200.2** remote-as 64512

#==={for **Tenant2 = AS 64513**}===
#neighbor **10.10.200.1** remote-as 64513
#neighbor **10.10.200.2** remote-as 64513

Node 1

Configure each Nebula Node for BGP Config

NOTE: Edit the /etc/quagga/bgpd.conf file on Node1 to look like the following...

```
#=================================================================
# Quagga BGPd configuration file for node1.
#=================================================================
hostname node1
password ciab2019
enable password ciab2019

#=================================================================
# bgp multiple-instances...
#=================================================================
# Define Autonomous System (AS) number for BGP
# example:  64512 Tenant1 and 64513 Tenant2
#
# NOTE: there can be more than 1 if multi-tenant -or-
# multi-network configuration that Node2 can belong to.

#=================================================================
router bgp 64512     # Tenant1
router bgp 64513     # Tenant2

#=================================================================
# ID "this" BGP Node (Node2) by its Nebula TUN (Nebula1) Interface IP address
#=================================================================

bgp router-id 10.10.200.1

#=================================================================
# Add Node1 as a BGP Peer node in AS 64512 for Tenant1 and AS 64513 for Tenant2
# to node Node2.
#
# One entry for each LXD Host/Server node using the its Nebula TUN (nebula1)
# interface IP and  AS Number that this Node2 should also belong to.
#
# NOTE:
# again, if this is a Multi-Tenant or a Multi-Network
# configuration EACH of these "could" be in a different AS
# Number which would effectively separate Traffic like a VPN
# would do except this would also Route via BGP.
#
# THIS Node would have to have more than one single
# "router bgp <AS number>" statement though so it'd know to
```

```
# participate in more than one AS Network
#============================================================
neighbor 10.10.200.2 remote-as 64512
neighbor 10.10.200.2 remote-as 64513

#============================================================
# Tell BGP to Share our local LXD subnet route info
#============================================================
network 10.216.160.0/24      # IP of Tenant1 custom LXD bridge
network 10.216.161.0/24      # IP of Tenant2 custom LXD bridge

#============================================================
# Define this Node1's BGP log file & location
#============================================================
log file /var/log/quagga/bgpd.log
log stdout
```

Node 2

BGP Configuration

NOTE: Edit the /etc/quagga/bgpd.conf file on Node2 to look like the following...

```
#=================================================================
# Quagga BGPd configuration file for node2.
#=================================================================

hostname node2
password ciab2020
enable password ciab2020

#=================================================================
# bgp multiple-instances...
#=================================================================
# Define Autonomous System (AS) number for BGP
# example:   64512 for Tenant1 and 64513 for Tenant2
#
# NOTE:  there can be more than 1 if multi-tenant -or-
#       multi-network configuration that Node2 can belong to.
#=================================================================

router bgp 64512
router bgp 64513

#=================================================================
# ID this BGP Node (Node2) by its Host's Nebula TUN (nebula1) Interface IP address
#=================================================================

bgp router-id 10.10.200.2

#=================================================================
# Add Node2 as a BGP Peer node for Tenant1 (AS 64512) and Tenant2 (AS 64513)
# to node Node1.
#
# One entry for each LXD Host/Server node using the its IP and AS Number that this
# Node2 should also belong to.
#
# NOTE: again, if this is a Multi-Tenant or a Multi-Network configuration EACH of
#           these "could" be in a different AS Number which would effectively separate
#           Traffic like a VPN would do except this would also Route via BGP.
#
#           THIS Node would have to have more than 1 "router bgp <AS number>"
#           statement though so it'd know to participate in more than one AS Network
#=================================================================
```

```
neighbor 10.10.200.1 remote-as 64512
neighbor 10.10.200.1 remote-as 64513

#=============================================================
# Tell BGP to Share our local LXD subnet route info
#=============================================================

network 10.87.33.0/24           # Tenant1 LXD custom Bridge IP
network 10.87.34.0/24           # Tenant2 LXD custom Bridge IP

#=============================================================
# Define this Node1's BGP log file & location
#=============================================================

log file /var/log/quagga/bgpd.log

log stdout
```

Step 3

On each Nebula Node Start Quagga & BGP Routing

Now that the configuration of Quagga and BGP for the Nebula Lighthouse Node and non-Lighthouse Nodes is complete you need to *start Quagga and BGP Routing on all Nebula Nodes including the Lighthouse Node:*

*# On **each** Node…*

$ sudo service quagga start

Step 4

Verification that BGP Routing is Working on each Node

The Quagga Shell

Quagga provides a shell called “vtysh” to interact with the quagga Routing daemon.

There are a many commands you can use to interact with the Quagga shell.

For our “verification”, on each Nebula node execute the following

$ vytsh -c 'show ip bgp'

'show ip bgp' gives you an listing of current BGP Peers and Shared routes on each Node.

$ vytsh -c 'show ip route'

'show ip route' gives you an listing of the Node’s Routing table.

$ vytsh -c 'show bgp neighbor'

'show ip bgp neighbor' gives you nformation on a Node’s BGP Peers and Connnections.

*Note: You can either run vtysh to enter the Quagga Shell and then type the commands or you can execute the Quagga shell directly from your shell using “**vtysh -c <command>**”.*

If you do not see a Node’s BGP Peers in the “***show ip bgp***” command recheck the Node’s bgpd.conf configuration and restart the quagga service.