ipinfusion™

QUICK START GUIDE

Running OcNOS® VMs in GNS3 Quick Start Guide

January 2023

Contents

About the OcNOS VM

The OcNOS Virtual Machine (VM) from IP Infusion helps you get familiar with OcNOS. The OcNOS VM runs on a standard x86 environment. The OcNOS VM is used to validate configurations and test L2, L3, and MPLS features at your own pace, with no costs associated. Without bare metal switches, OcNOS VM can be run on popular environments like GNS3 and hypervisors including KVM, VirtualBox, and VMware. This document provides information on how to run OcNOS VM in the GNS3 environment.

All basic Layer 2, Layer 3, and multicast functionality are available. MPLS support is also available, including limited support of MPLS forwarding. The OcNOS VM comes with a 365 days valid license.

The data plane forwarding functions have limited support. OcNOS VM is designed for feature testing, and not for data plane performance testing or full bandwidth traffic testing.

Benefits of the OcNOS VM

Following are benefits of OcNOS VM:

- Free
- No need to wait for the hardware
- Get familiar with OcNOS software
- Validate configurations
- Test L2, L3, and MPLS features without any risk
- Prototype network operations

Feature List

CLIs for the following features are available. The complete feature set of OcNOS is supported on Commercial off-the-shelf (COTS) hardware platforms switches from Dell, Delta Agema, Edgecore, and UFISpace. For the complete feature list, please contact IP Infusion Sales.

SYSTEM FEATURES

- ARP support
- SSH/Telnet
- SNMP
- Debugging and logging
- AAA
- DHCP, DNS

LAYER-2 FEATURES

- STP/RSTP/MSTP
- BPDU Guard and Root Guard
- VLAN, Private VLAN
- LACP
- LLDP
- VLAN Interface
- QinQ
- 802.1x

LAYER-3 FEATURES

- IPv4 Routing
- VRF Support
- RIP v2, RIP NG
- BFD with BGP, OSPF, ISIS
- BGP
- OSPF v2, OSPF v3
- ISIS
- VRRP

MPLS FEATURES

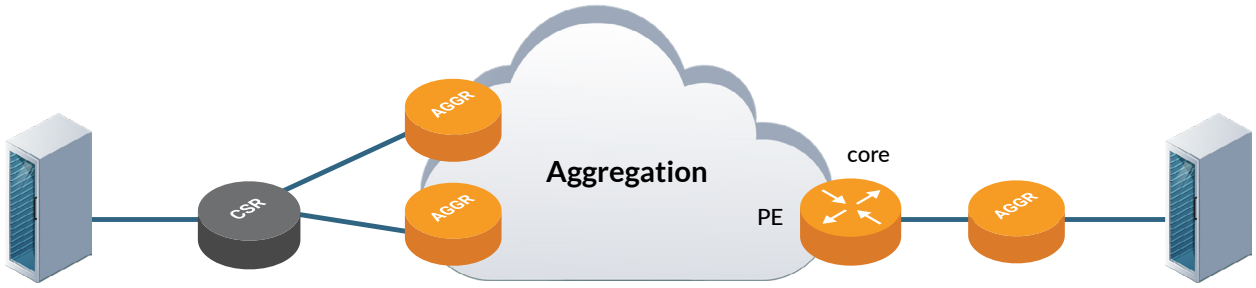
- MPLS Label Switching
- LDP and RSVP Support
- RSVP FRR
- VPLS with LDP Signaling
- VPWS with 1:1 backup support
- BGP MPLS L3VPN
- MPLS DCI using ICCP and VPLS redundancy

MULTICAST FEATURES

- IGMP
- PIM-SM/SSM/DM
- MSDP Support

Running OcNOS in GNS3

This section describes how to install GNS3 in VMware hypervisor and run OcNOS VM switches and test servers in GNS3 environment. We will create following switch topology shown below to test OcNOS L2 and L3 software features. In this example, we will test the BGP and L3 VPN feature. The following is a test topology in a GNS3 environment.



One Cell Site Router (CSR), three Aggregation Routers (AGGR) and a core router are used in this GNS3 test topology. Two Debian Linux servers are used in GNS3 for generating the test traffic.

System Requirements for Running OcNOS VMs in GNS3

Following system requirements are used for running OcNOS VMs in GNS3. We will run GNS3 as a remote server VM in the VMware hypervisor. Following are requirements for running a remote GNS3 server VM:

- VMware vSphere Hypervisor (ESXi) 6.5.0 or later
- VM requirements:
  - CPU: 2 vCPUs. CPU need to support the nested VM in the ESXi server for running GNS3. Please refer to the next section for details.
  - Memory: 16 GB
  - Hard Disk: 40 GB
  - NICs: 1. Make sure there is a DHCP server on the network this NIC card is connected to.
- We will be using GNS3 project image that contains the following VMs: five OcNOS VMs (version 6.0.2) with BGP and L3 VPN configuration, and 2 Debian Linux Servers.

Files Provided for Running OcNOS VMs in GNS3

Following files are provided for running OcNOS VMs in GNS3:

1. ***BGP-L3VPN-Proj.gns3project*** file includes OcNOS VM image, OcNOS VM GNS3 Template and includes configuration for all switches and servers for BGP and L3 VPN test case. Use this file to populate BGP and L3 VPN topology and test in the GNS3 environment. It will get you up and running quickly. This is the only file you need for the above purpose.
2. Alternatively, if you want to create your own topology in GNS3 and test, use the following two files:
   a. ***DEMO_VM-OcNOS-6.0.2.11-MPLS-x86-MR.qcow2.xz***: This is OcNOS VM image for the GNS3 environment. OcNOS VM image file is archive compressed using XZ compression. Use Mac OS *Archive Utility* or *7-zip tools* to uncompress the file. To uncompress the file in Linux, use the command *xz -d <file_name>.xz*
   b. ***OCNOS.gns3a***: This is OcNOS QEMU VM Template. You can import this template to create OcNOS VMs in GNS3.

Setup the GNS3 Environment for Validating BGP and L3 VPN

Setting up above topology in GNS3 for validating BGP and L3 VPN requires the following four steps:

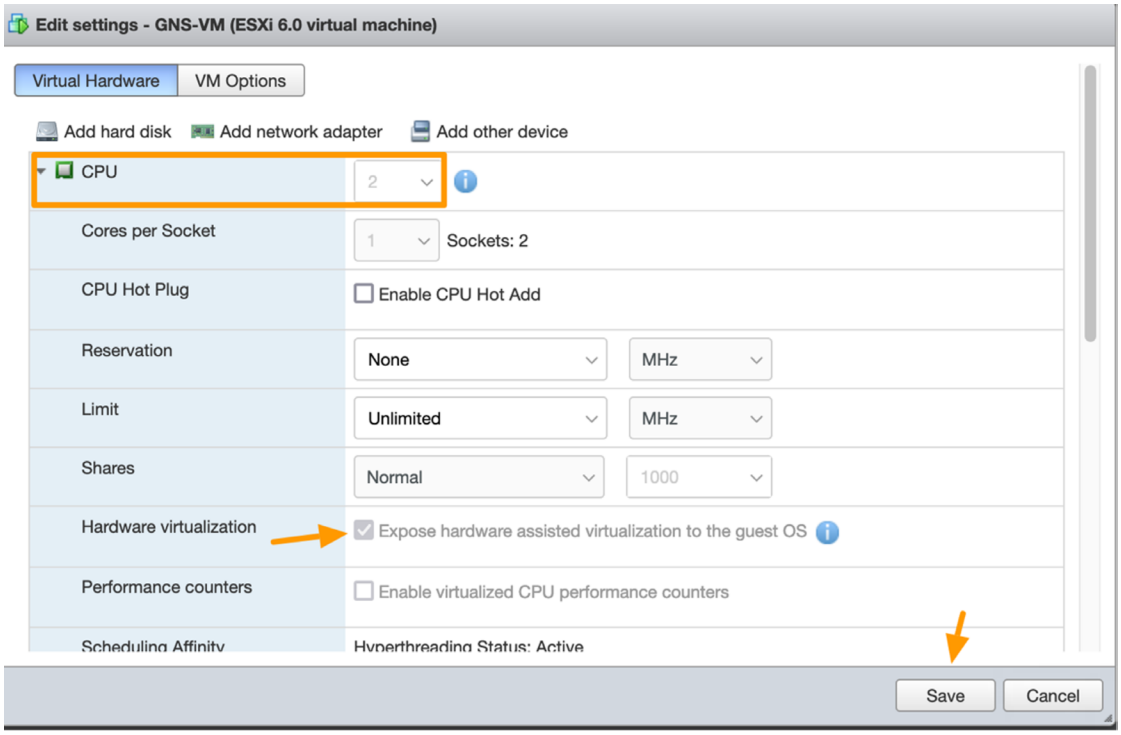
1. Install the remote GNS3 server VM in the VMware hypervisor
2. Install and configure GNS3 client in your MacOS or Windows laptop
3. Import an example GNS3 Project called *BGP and L3 VPN* Project on the remote GNS3 VM
4. Verify *BGP and L3 VPN* Project

1. Install the Remote GNS3 Server VM in the VMware vSphere Hypervisor

The following are steps to install a GNS3 VM in a VMware vSphere hypervisor:

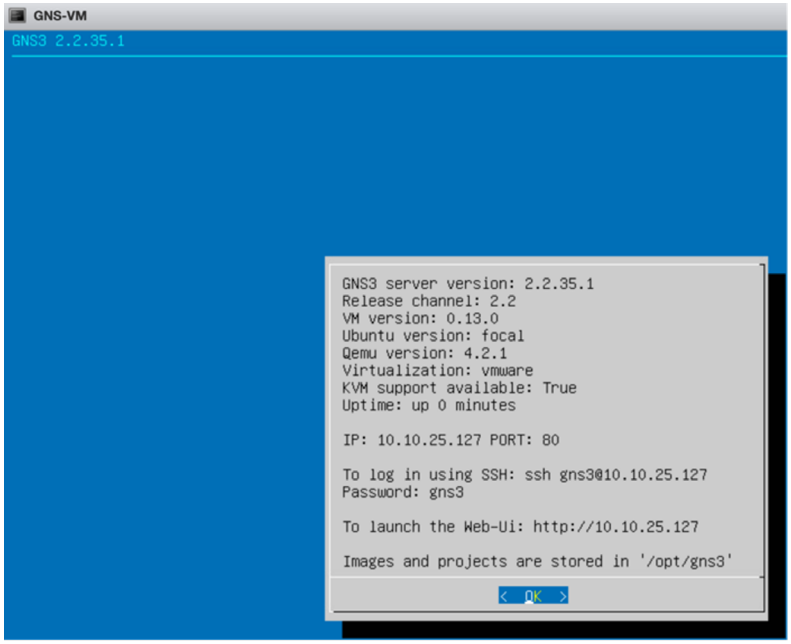
a. **Download GNS3 VM** to run in VMware vSphere ESXi hypervisor. In this example GSN3 VM version 2.2.35.1 and VMware ESXi version 7.0.3 are used for testing.

b. Install the GNS3 VM on the ESXi server: Follow the documentation given in this link to install GNS3. After you install the GNS3 VM, turn off the VM power, select edit settings and expand CPU to check the nested VM support in the ESXi server. Hardware Virtualization needs to be enabled in this case as shown below. Set the CPU to 2.



In addition set the Memory of the VM to 16 GB and click Save. VM takes up 18Gb of hard disk space.

c. Power up the VM and click OK on the VM console as shown below.



The GNS3 VM gets its IP address 10.10.25.127 and TCP port 80 in this example from the DHCP server and the IP address is displayed on the console. The credentials for login are also given in the console: username is gns3 and password is gns3. The Web URL to access the GNS environment is given as http://10.10.25.127.

2. Install and Configure GNS3 Client

Following are steps to install and configure GNS3 client on your laptop and setup a GNS3 environment for testing OcNOS features:

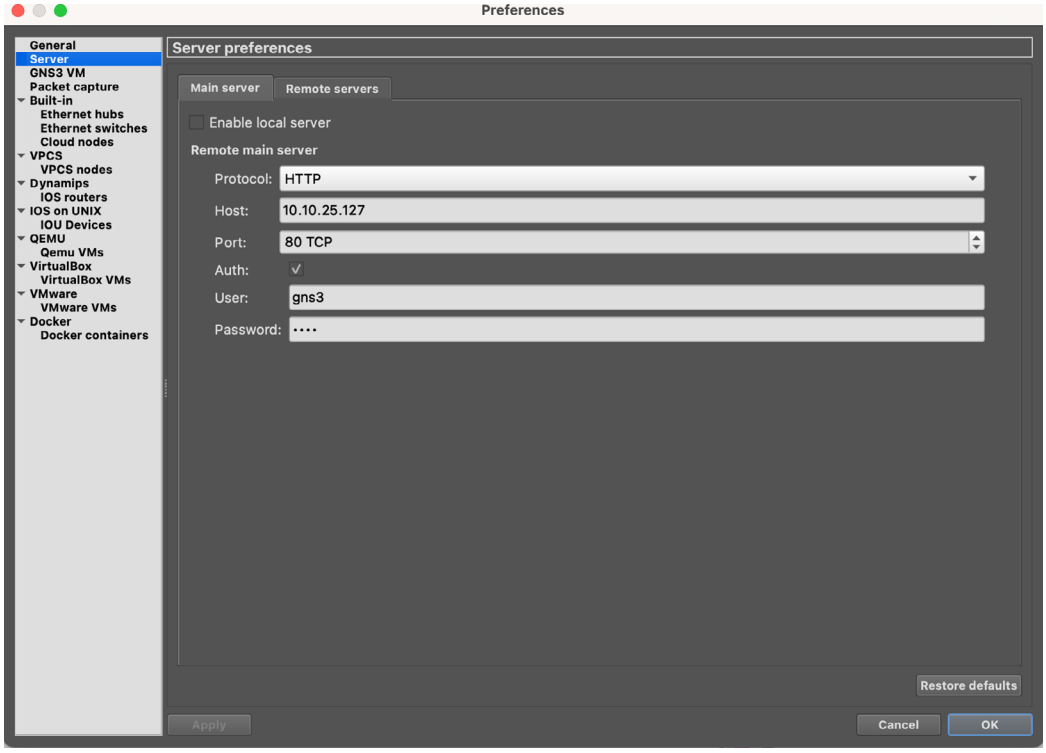
To install GNS3 client on your laptop, follow the instructions provided below:

a. **Install GNS3 client on the laptop**: Use the following link for MacOS: Install GNS3 client on a Mac OS X laptop. In this example GNS3 client version 2.2.35.1 is used. Please note GNS3 server version need to match with GNS3 client version. Use the following link for Windows: Install GNS3 client on a Windows laptop.

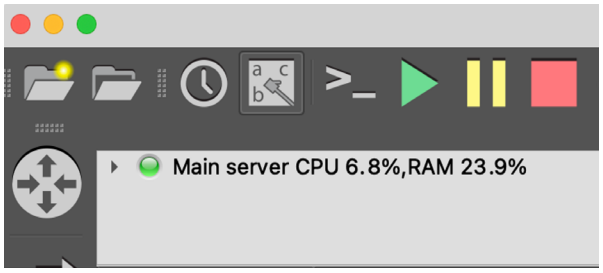
b. **Run GNS3 client application** on your laptop.

c. **Configure the GNS3 server** in the GNS3 Application as follows. Select *GNS3->Preferences and select Main server* tab enter GNS server parameters as shown below (use GNS3 VM info) and click Apply and OK. Please note: *Enable local server* should not be selected.

d.



Verify the Main server configuration



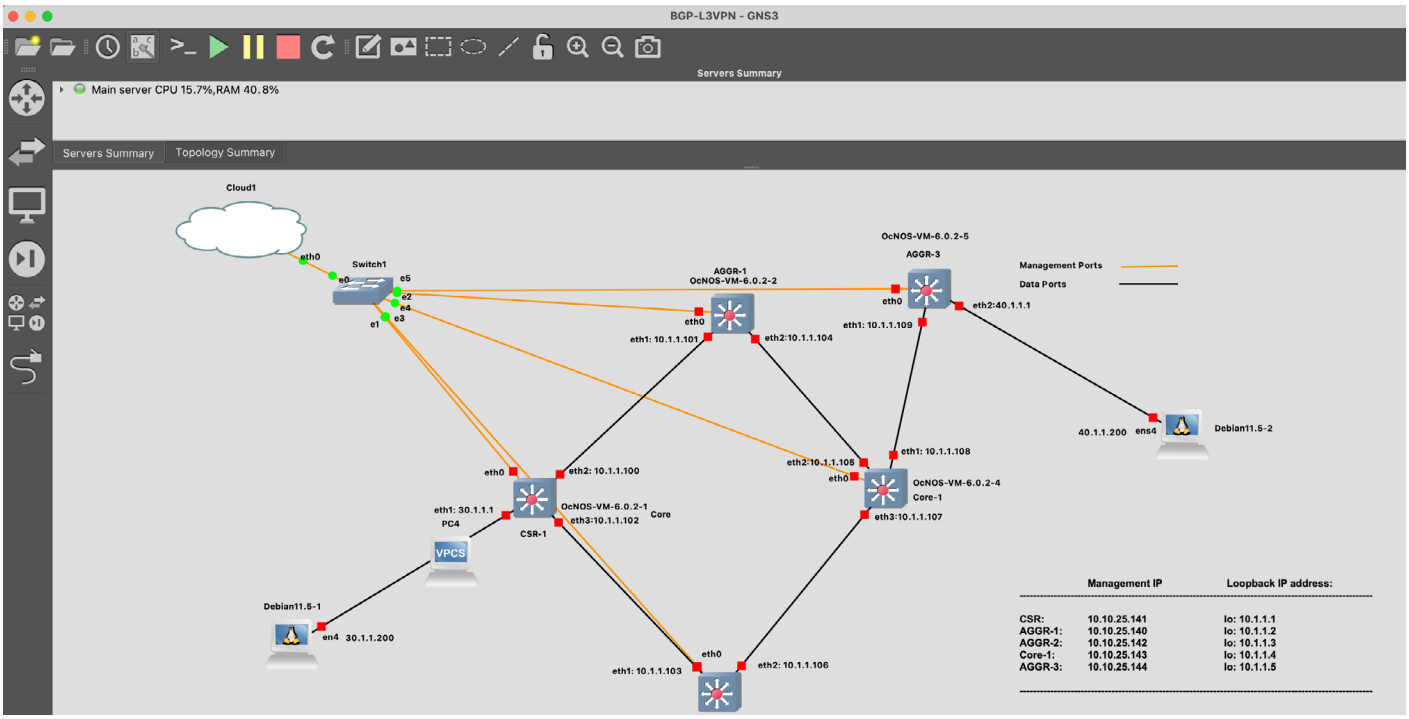
Select *Server Summary* tab, you will see the Main Server running with green color status.

3. Import an Example GNS3 Project Called *BGP and L3 VPN* on the Remote GNS3 VM

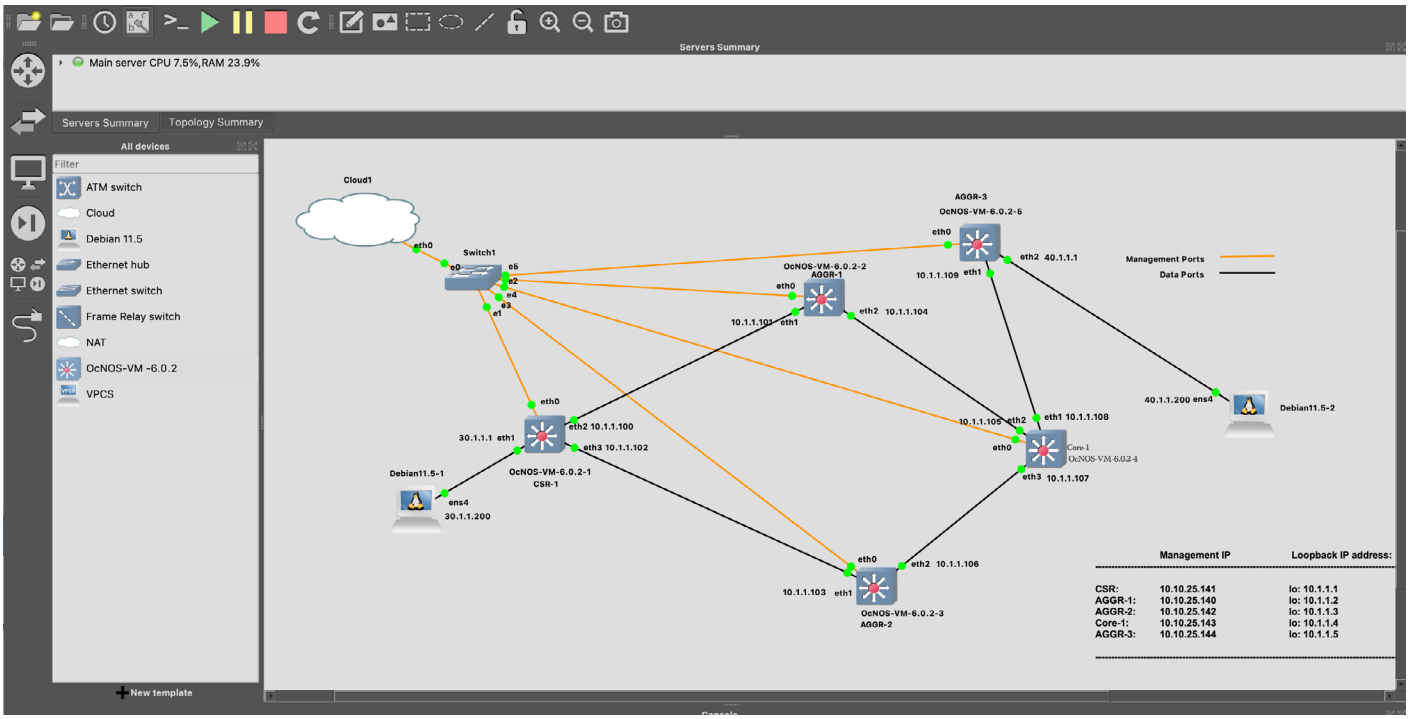
BGP-L3 VPN project file ***BGP-L3VPN-Proj.gns3project*** includes OcNOS VM image, OcNOS VM GNS3 Template and includes configuration for all switches and servers for BGP and L3 VPN test case. For future use you can feel free to change the topology or change configuration in any of the OcNOS switches.

Download the BGP-L3VPN.gns3project file to your laptop from the IP Infusion website. Select *File* > *Import portable project* menu and select the gns3 project file you have downloaded earlier and give it a project name as shown below:

Click *OK*. Press *Play* button at the top to start all devices in GNS3 project.



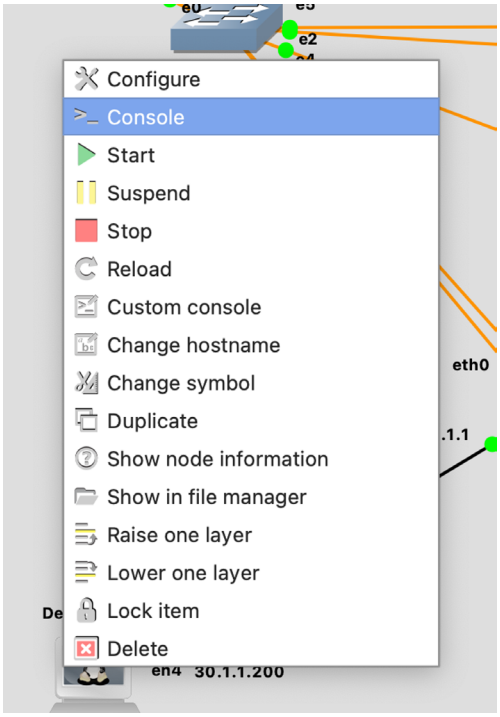
Once all devices are up you will see the following the following screen with all ports in green status:



Please note: Management IP addresses for your OcNOS switches will be different than the ones given in the above screen. You need to edit the above screen to change Management IP addresses. To get the Management IP address of a specific switch, please use the info provided in the section. To edit the screen to change the Management IP address, click the Pencil Icon at the top menu.

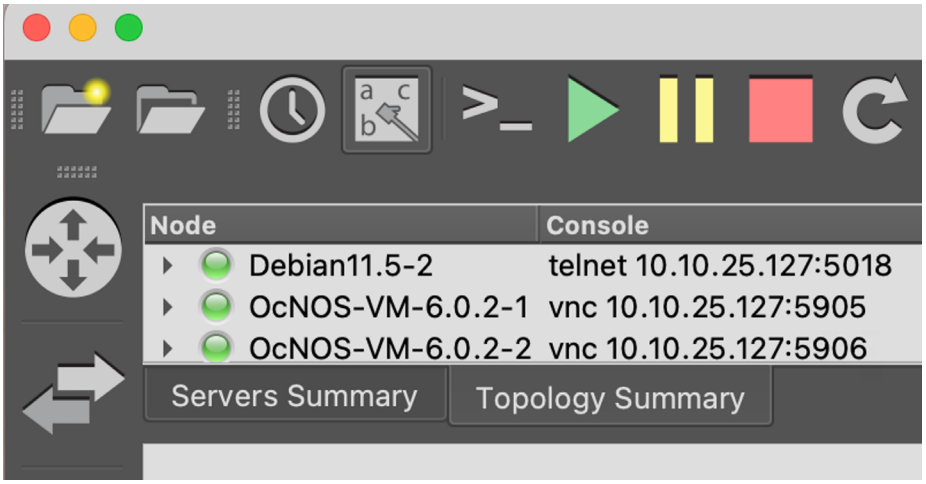
GET FAMILIAR WITH ACCESSING SERVERS AND SWITCHES:

a. To access the server, right click on the server and select console.



Following are default credentials to log into the Debian Server: debian/debian
From the server Linux CLI prompt you can run CLI commands to generate test traffic.

b. To access one of the OcNOS VM switch, check the Topology summary tab as shown below in the GNS3 Application:



Select the *Topology Summary* tab and note down the IP address of the connections shown above: Using VNC application connect to the specific OcNOS switch with its corresponding address (10.10.25.127:5905 in the above example): Default switch login credentials are *ocnos/ocnos*. Run the following commands to check the IP address of the Management Port *eth0*.

Now on you can SSH to the switch using the following command from your laptop CLI: *ssh ocnos@10.10.25.141* and enter password as *ocnos*. You can use SSH session for running verification commands listed in later section.

4. Verify *BGP and L3 VPN* Project

We will run several commands to verify BGP and L3 VPN functionalities.

a. **Generate Test Traffic**: Log into the console of the Debian11.2-1 Server from the GNS3 client Application (by right clicking on the server and choose Console) and execute the following Linux shell command to send 1000 packets from the Debian11.2-1 Server to the Debian11.2-2 Server on the TEST_VRF.

```
debian@debian:~$ ping -c 1000 -i 1 40.1.1.200
PING 40.1.1.200 (40.1.1.200) 56(84) bytes of data.
64 bytes from 40.1.1.200: icmp_seq=1 ttl=63 time=4.15 ms
64 bytes from 40.1.1.200: icmp_seq=2 ttl=63 time=4.84 ms
64 bytes from 40.1.1.200: icmp_seq=3 ttl=63 time=5.45 ms
64 bytes from 40.1.1.200: icmp_seq=4 ttl=63 time=3.56 ms
64 bytes from 40.1.1.200: icmp_seq=5 ttl=63 time=3.63 ms
...
```

b. **Check summary of known neighbor**: Log into the console of the CSR-1 OcNOS virtual switch (or SSH into CSR-1) and run the following commands to verify the BGP and L3 VPN functionalities. The show clns neighbors command provides a summary of known neighbors, the connecting interface, and the state of the adjacency.

```
CSR-1#show clns neighbors

Total number of L1 adjacencies: 2
Total number of L2 adjacencies: 0
Total number of adjacencies: 2
Tag 1:  VRF : default
System Id      Interface   SNPA            State  Holdtime  Type Protocol
AGGR-1         eth2        0cc6.74db.0001  Up     6         L1   IS-IS
AGGR-2         eth3        0c2c.0e08.0001  Up     27        L1   IS-IS
```

c. **Check TEST_VRF forwarding table**: Following output shows we have path to reach the second server.

```
CSR-1# show mpls vrf-forwarding-table vrf TEST_VRF
Owner    FEC            FTN-ID   Oper-Status   Out-Label   Tunnel-id   NHLFE-id   Out-Intf   Nexthop
BGP      40.1.1.0/24    1        Up    24320       0           7          eth2       10.1.1.5
```

Also check Incoming Label Map entries. Use the following command to view Incoming label mapping (ILM) table entries

```
CSR-1#show mpls ilm-table
Codes: > - installed ILM, * - selected ILM, p - stale ILM
        K - CLI ILM, T - MPLS-TP, s - Stitched ILM
       S - SNMP, L - LDP, R - RSVP, C - CRLDP
       B - BGP , K - CLI , V - LDP_VC, I - IGP_SHORTCUT
       O - OSPF/OSPF6 SR, i - ISIS SR, k - SR CLI
       P - SR Policy, U - unknown

Code     FEC/VRF/L2CKT     ILM-ID      In-Label     Out-Label    In-Intf     Out-Intf/VRF       Nexthop
LSP-Type
   L>    10.1.1.106/31   11    24965       3           N/A        eth3      10.1.1.103          LSP_DEFAULT
   L>    10.1.1.3/32     7     24961       3           N/A        eth3      10.1.1.103          LSP_DEFAULT
   B>    TEST_VRF        1     24320       Nolabel     N/A        TEST_VRF     N/A              LSP_DEFAULT
   L>    10.1.1.2/32     13    24967       3           N/A        eth2      10.1.1.101          LSP_DEFAULT
   L>    10.1.1.104/31   14    24968       3           N/A        eth2      10.1.1.101          LSP_DEFAULT
```

d. **Check for path to AGGR-3 in MPLS forwarding Table**: Run the following command in CSR-1.

```
CSR-1#show mpls forwarding-table
Codes: > - installed FTN, * - selected FTN, p - stale FTN,
       B - BGP FTN, K - CLI FTN, t - tunnel, P - SR Policy FTN,
       L - LDP FTN, R - RSVP-TE FTN, S - SNMP FTN, I - IGP-Shortcut,
       U - unknown FTN, O - SR-OSPF FTN, i - SR-ISIS FTN, k - SR-CLI FTN

Code     FEC                FTN-ID    Nhlfe-ID  Tunnel-id   Pri   LSP-Type     Out-Label    Out-Intf     ELC    Nexthop
   L>    10.1.1.2/32        1         32        -           Yes   LSP_DEFAULT  3            eth2         No
10.1.1.101
   L>    10.1.1.3/32        2         14        -           Yes   LSP_DEFAULT  3            eth3         No
10.1.1.103
   L>    10.1.1.4/32        3         16        -           Yes   LSP_DEFAULT  24962        eth3         No
10.1.1.103
                                      48        -           Yes   LSP_DEFAULT  24962        eth2         No
10.1.1.101
   L>    10.1.1.5/32        4         20        -           Yes   LSP_DEFAULT  24963        eth3         No
10.1.1.103
                                      49        -           Yes   LSP_DEFAULT  24963        eth2         No
10.1.1.101
   L>    10.1.1.104/31      5         32        -           Yes   LSP_DEFAULT  3            eth2         No
10.1.1.101
   L>    10.1.1.106/31      6         14        -           Yes   LSP_DEFAULT  3            eth3         No
10.1.1.103
   L>    10.1.1.108/31      7         28        -           Yes   LSP_DEFAULT  24965        eth3         No
10.1.1.103
                                      50        -           Yes   LSP_DEFAULT  24966        eth2         No    10.1.1.101
```

You can see AGGR-5 can be reached via eth2 and eth3.

e. **Check LDP sessions in CSR-1:** `Execute the following CLI in CSR-1.`

```
CSR-1#show ldp session
Peer IP Address              IF Name     My Role     State       KeepAlive  UpTime
10.1.1.2                     eth2        Passive    OPERATIONAL    30     22:24:09
10.1.1.3                     eth3        Passive    OPERATIONAL    30     22:24:09
```

f. **Check route between two Debian Servers:** Check the route from one Debian Server to other using the following command:
One server is directly connected to 30.1.1.0/24 network and other server in 40.1.1.0/24 network is accessible via BGP.

```
CSR-1#show ip route vrf TEST_VRF database
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2,
       ia - IS-IS inter area, E - EVPN,
       v - vrf leaked
       > - selected route, * - FIB route, p - stale info

IP Route Table for VRF "TEST_VRF"
C    *>  30.1.1.0/24 is directly connected, eth1, 1d07h49m
B    *>  40.1.1.0/24 [200/0] via 10.1.1.5, 00:20:26

Gateway of last resort is not set
```

g. **Check L3VPN routes:** Use the following command to display information relating to MPLS VPN.

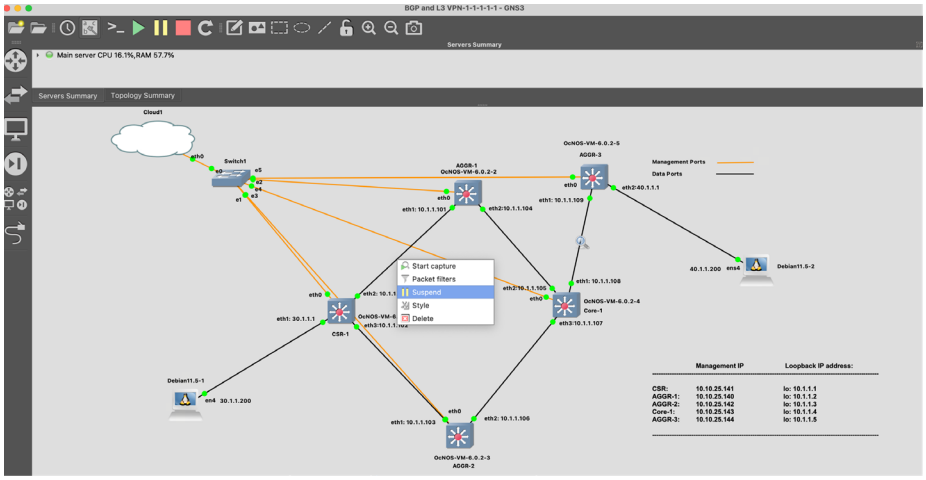
```
CSR-1#show ip bgp vpnv4 all summary
BGP router identifier 10.1.1.1, local AS number 65000
BGP table version is 9
1 BGP AS-PATH entries
0 BGP community entries

Neighbor                V   AS   MsgRcv    MsgSen TblVer   InQ   OutQ    Up/Down   State/PfxRcd
10.1.1.2                4 65000 4446       4444       9     0      0  00:20:32              1
10.1.1.3                4 65000 4420       4418       9     0      0  00:20:37              1

Total number of neighbors 2
Total number of Established sessions 2
```

h. **Stop flow of traffic between CSR-1 and AGGR-1 and verify whether traffic flows from one server to the other:**

When the ICMP traffic is flowing, right click on the link between the ***CSR-1*** and the ***AGGR-1***. Select ***Suspend*** to stop the traffic flowing through that link as shown below. Now traffic will not go through *eth2* interface. Traffic will only go through *eth3* interface.



Check the traffic flow using the following command in CSR-1.

```
CSR-1#show mpls forwarding-table
Codes: > - installed FTN, * - selected FTN, p - stale FTN,
       B - BGP FTN, K - CLI FTN, t - tunnel, P - SR Policy FTN,
       L - LDP FTN, R - RSVP-TE FTN, S - SNMP FTN, I - IGP-Shortcut,
       U - unknown FTN, O - SR-OSPF FTN, i - SR-ISIS FTN, k - SR-CLI FTN

Code    FEC              FTN-ID    Nhlfe-ID  Tunnel-id   Pri   LSP-Type      Out-Label   Out-Intf    ELC   Nexthop
   L>   10.1.1.2/32      1         10        -           Yes   LSP_DEFAULT   24961       eth3        No
10.1.1.103
   L>   10.1.1.3/32      2         14        -           Yes   LSP_DEFAULT   3           eth3        No
10.1.1.103
   L>   10.1.1.4/32      3         16        -           Yes   LSP_DEFAULT   24962       eth3        No
10.1.1.103
   L>   10.1.1.5/32      4         20        -           Yes   LSP_DEFAULT   24963       eth3        No
10.1.1.103
   L>   10.1.1.104/31    5         27        -           Yes   LSP_DEFAULT   24964       eth3        No
10.1.1.103
   L>   10.1.1.106/31    6         14        -           Yes   LSP_DEFAULT   3           eth3        No
10.1.1.103
   L>   10.1.1.108/31    7         28        -           Yes   LSP_DEFAULT   24965       eth3        No
10.1.1.103
```

i. Verify whether traffic can reach AGGR-3 with MPLS ping:

```
CSR-1#ping mpls ldp 10.1.1.5/32 detail
Sending 5 MPLS Echos to 10.1.1.5, timeout is 5 seconds

Codes:
'!' - Success, 'Q' - request not sent, '.' - timeout,
'x' - Retcode 0, 'M' - Malformed Request, 'm' - Errored TLV,
'N' - LBL Mapping Err, 'D' - DS Mismatch,
'U' - Unknown Interface, 'R' - Transit (LBL Switched),
'B' - IP Forwarded, 'F' No FEC Found, 'f' - FEC Mismatch,
'P' - Protocol Error, 'X' - Unknown code,
'Z' - Reverse FEC Validation Failed

 Type 'Ctrl+C' to abort

! seq_num = 1 10.1.1.109 1.92 ms
! seq_num = 2 10.1.1.109 1.01 ms
! seq_num = 3 10.1.1.109 1.26 ms
! seq_num = 4 10.1.1.109 1.63 ms
! seq_num = 5 10.1.1.109 2.52 ms

Success Rate is 100.00 percent (5/5)
round-trip min/avg/max = 1.01/1.77/2.52
```

References

OcNOS

The following are reference materials related to OcNOS:

- OcNOS Configuration Guides

GNS3

The following are reference materials related to GNS3:

- Getting Started with GNS3

Appendix-A - Example *BGP and L3 VPN* Configuration Used in the GNS3 Environment

The following example configurations are used in the GNS3 environment to test BGP and L3 VPN functionality in OcNOS virtual switches.

CSR-1 Switch Configuration

The configuration used in the CSR-1 OcNOS virtual switch is given below:

```
!
no service password-encryption
!
logging console 2
logging monitor 7
logging cli
!
ip vrf management
!
ip vrf TEST_VRF
   rd 10.1.1.1:1
   route-target both 65000:1
!
hostname CSR-1
ip domain-lookup
feature telnet
feature ssh
feature rsyslog
!
router ldp
   router-id 10.1.1.1
   transport-address ipv4 10.1.1.1
!
!
interface lo
   ip address 127.0.0.1/8
   ip address 10.1.1.1/32 secondary
   ipv6 address ::1/128
   ip router isis 1
!
interface eth0
   ip vrf forwarding management
   ip address dhcp
!
interface eth1
   ip vrf forwarding TEST_VRF
   ip address 30.1.1.1/24
!
interface eth2
   ip address 10.1.1.100/31
   label-switching
   mpls ldp-igp sync isis level-1
```

```
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth3
   ip address 10.1.1.102/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth4
!
exit
!
router isis 1
   is-type level-1
   metric-style wide level-1
   mpls traffic-eng router-id 10.1.1.1
   mpls traffic-eng level-1
   capability cspf
   dynamic-hostname
   bfd all-interfaces
   net 49.0111.1100.0075.0001.00

!
router bgp 65000
   bgp router-id 10.1.1.1
   neighbor 10.1.1.2 remote-as 65000
   neighbor 10.1.1.3 remote-as 65000
   neighbor 10.1.1.2 update-source lo
   neighbor 10.1.1.3 update-source lo
!
address-family vpnv4 unicast
   neighbor 10.1.1.2 activate
   neighbor 10.1.1.3 activate
   exit-address-family
!
address-family ipv4 vrf  TEST_VRF
redistribute connected
exit-address-family
!
line vty 0
   exec-timeout 0 0
!
!
end
```

AGGR-1 Switch Configuration

The configuration used in the AGGR-1 OcNOS virtual switch is given below:

```
!
no service password-encryption
!
logging console 2
logging monitor 7
logging cli
!
ip vrf management
!
hostname AGGR-1
no ip domain-lookup
ip domain-lookup vrf management
feature telnet vrf management
feature ssh vrf management
feature rsyslog vrf management
!
router ldp
   router-id 10.1.1.2
   transport-address ipv4 10.1.1.2
!
!
interface lo
   ip address 127.0.0.1/8
   ip address 10.1.1.2/32 secondary
   ipv6 address ::1/128
   ip router isis 1
!
interface eth0
   ip vrf forwarding management
   ip address dhcp
!
interface eth1
   ip address 10.1.1.101/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth2
   ip address 10.1.1.104/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
```

```
   set lldp enable txrx
   exit
!
interface eth3
!
interface eth4
!
exit
!
router isis 1
   is-type level-1
   metric-style wide level-1
   mpls traffic-eng router-id 10.1.1.2
   mpls traffic-eng level-1
   capability cspf
   dynamic-hostname
   bfd all-interfaces
   net 49.0111.1100.0075.0002.00
!
router bgp 65000
   no bgp inbound-route-filter
   bgp router-id 10.1.1.2
   neighbor 10.1.1.1 remote-as 65000
   neighbor 10.1.1.3 remote-as 65000
   neighbor 10.1.1.4 remote-as 65000
   neighbor 10.1.1.5 remote-as 65000
   neighbor 10.1.1.1 update-source lo
   neighbor 10.1.1.3 update-source lo
   neighbor 10.1.1.4 update-source lo
   neighbor 10.1.1.5 update-source lo
!
address-family vpnv4 unicast
   neighbor 10.1.1.1 activate
   neighbor 10.1.1.1 route-reflector-client
   neighbor 10.1.1.3 activate
   neighbor 10.1.1.4 activate
   neighbor 10.1.1.4 route-reflector-client
   neighbor 10.1.1.5 activate
   neighbor 10.1.1.5 route-reflector-client
   exit-address-family
!
line vty 0
   exec-timeout 0 0
!
!
end
```

AGGR-2 Switch Configuration

The configuration used in the AGGR-2 OcNOS virtual switch is given below:

```
!
no service password-encryption
!
logging console 2
logging monitor 7
logging cli
!
ip vrf management
!
hostname AGGR-2
no ip domain-lookup
ip domain-lookup vrf management
feature telnet vrf management
feature ssh vrf management
feature rsyslog vrf management
!
router ldp
   router-id 10.1.1.3
   transport-address ipv4 10.1.1.3
!
!
interface lo
   ip address 127.0.0.1/8
   ip address 10.1.1.3/32 secondary
   ipv6 address ::1/128
   ip router isis 1
!
interface eth0
   ip vrf forwarding management
   ip address dhcp
!
interface eth1
   ip address 10.1.1.103/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth2
   ip address 10.1.1.106/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
```

```
   set lldp enable txrx
   exit
!
interface eth3
!
interface eth4
!
exit
!
router isis 1
   is-type level-1
   metric-style wide level-1
   mpls traffic-eng router-id 10.1.1.3
   mpls traffic-eng level-1
   capability cspf
   dynamic-hostname
   bfd all-interfaces
   net 49.0111.1100.0075.0003.00
!
router bgp 65000
   bgp router-id 10.1.1.3
   no bgp inbound-route-filter
   neighbor 10.1.1.1 remote-as 65000
   neighbor 10.1.1.2 remote-as 65000
   neighbor 10.1.1.4 remote-as 65000
   neighbor 10.1.1.5 remote-as 65000
   neighbor 10.1.1.1 update-source lo
   neighbor 10.1.1.2 update-source lo
   neighbor 10.1.1.4 update-source lo
   neighbor 10.1.1.5 update-source lo
!
address-family vpnv4 unicast
   neighbor 10.1.1.1 activate
   neighbor 10.1.1.1 route-reflector-client
   neighbor 10.1.1.2 activate
   neighbor 10.1.1.2 route-reflector-client
   neighbor 10.1.1.4 activate
   neighbor 10.1.1.4 route-reflector-client
   neighbor 10.1.1.5 activate
   neighbor 10.1.1.5 route-reflector-client
   exit-address-family
!
line vty 0
   exec-timeout 0 0
!
!
end
```

CORE-1 Switch Configuration

The configuration used in the CORE-1 OcNOS virtual switch is given below:

```
no service password-encryption
!
logging console 2
logging monitor 7
logging cli
!
ip vrf management
!
hostname core-1
no ip domain-lookup
ip domain-lookup vrf management
feature telnet vrf management
feature ssh vrf management
feature rsyslog vrf management
!
router ldp
   router-id 10.1.1.4
   transport-address ipv4 10.1.1.4
!
interface lo
   ip address 127.0.0.1/8
   ip address 10.1.1.4/32 secondary
   ipv6 address ::1/128
   ip router isis 1
!
interface eth0
   ip vrf forwarding management
   ip address dhcp
!
interface eth1
   ip address 10.1.1.108/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth2
   ip address 10.1.1.105/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
```

```
!
interface eth3
   ip address 10.1.1.107/31
   label-switching
   mpls ldp-igp sync isis level-1
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth4
!
exit
!
router isis 1
   is-type level-1
   metric-style wide level-1
   mpls traffic-eng router-id 10.1.1.4
   mpls traffic-eng level-1
   capability cspf
   dynamic-hostname
   bfd all-interfaces
   net 49.0111.1100.0075.0004.00
!
router bgp 65000
   bgp router-id 10.1.1.4
   neighbor 10.1.1.2 remote-as 65000
   neighbor 10.1.1.3 remote-as 65000
   neighbor 10.1.1.5 remote-as 65000
   neighbor 10.1.1.2 update-source lo
   neighbor 10.1.1.3 update-source lo
   neighbor 10.1.1.5 update-source lo
!
address-family vpnv4 unicast
   neighbor 10.1.1.2 activate
   neighbor 10.1.1.3 activate
   neighbor 10.1.1.5 activate
   exit-address-family
!
line vty 0
   exec-timeout 0 0
!
!
end
```

AGGR-3 Switch Configuration

The configuration used in the AGGR-3 OcNOS virtual switch is given below:

```
!
no service password-encryption
!
logging console 2
logging monitor 7
logging cli
!
ip vrf management
!
ip vrf TEST_VRF
   rd 10.1.1.5:1
   route-target both 65000:1
!
hostname AGGR-3
ip domain-lookup
feature telnet
feature ssh
feature rsyslog
!
router ldp
   router-id 10.1.1.5
   transport-address ipv4 10.1.1.5
!
interface lo
   ip address 127.0.0.1/8
   ip address 10.1.1.5/32 secondary
   ipv6 address ::1/128
   ip router isis 1
!
interface eth0
   ip vrf forwarding management
   ip address dhcp
!
interface eth1
   ip address 10.1.1.109/31
   label-switching
   mpls ldp-igp sync isis level-1
   isis network point-to-point
   ip router isis 1
   enable-ldp ipv4
   lldp-agent
   set lldp enable txrx
   exit
!
interface eth2
   ip vrf forwarding TEST_VRF
   ip address 40.1.1.1/24
   lldp-agent
   set lldp enable txrx
   exit
!
```

```
interface eth3
!
interface eth4
!
exit
!
router isis 1
   is-type level-1
   metric-style wide level-1
   mpls traffic-eng router-id 10.1.1.5
   mpls traffic-eng level-1
   capability cspf
   dynamic-hostname
   bfd all-interfaces
   net 49.0111.1100.0075.0005.00
!
router bgp 65000
   bgp router-id 10.1.1.5
   neighbor 10.1.1.1 remote-as 65000
   neighbor 10.1.1.2 remote-as 65000
   neighbor 10.1.1.3 remote-as 65000
   neighbor 10.1.1.4 remote-as 65000
   neighbor 10.1.1.1 update-source lo
   neighbor 10.1.1.2 update-source lo
   neighbor 10.1.1.3 update-source lo
   neighbor 10.1.1.4 update-source lo
!
address-family vpnv4 unicast
   neighbor 10.1.1.1 activate
   neighbor 10.1.1.2 activate
   neighbor 10.1.1.3 activate
   neighbor 10.1.1.4 activate
   exit-address-family
!
address-family ipv4 vrf TEST_VRF
   redistribute connected
   exit-address-family
!
line vty 0
   exec-timeout 0 0
!
!
end
```

ABOUT IP INFUSION

IP Infusion is a leading provider of open network software and solutions for carriers, service providers and data center operators. Our solutions enable network operators to disaggregate their networks to accelerate innovation, streamline operations, and reduce Total Cost of Ownership (TCO). Network OEMs may also disaggregate network devices to expedite time to market, offer comprehensive services, and achieve carrier grade robustness. IP Infusion network software platforms have a proven track record in carrier-grade open networking with over 500 customers and over 10,000 deployments. IP Infusion is headquartered in Santa Clara, Calif., and is a wholly owned and independently operated subsidiary of ACCESS CO., LTD. Additional information can be found at **http://www.ipinfusion.com**

Phone | +1-877-699-3267 **Email |** sales@ipinfusion.com **Web |** www.ipinfusion.com