SONiC Deployment and Testing Using GNS3

Revision History

Revision No.	Description	Editor	Date
1.0	SONiC Deployment and Testing Using GNS3	Humza Altaf, Mati Ur Rehman, Hamza Iqbal, Muhammad Danish	May 10, 2023

Table of Contents

Introduction

Software for Open Networking in Cloud (SONiC) has emerged as a prominent open source network operating system, garnering widespread acceptance among hyperscalers, operators, and vendors. Its flexible architecture and robust feature set have made it a preferred choice for building scalable and efficient networks. This document aims to provide a comprehensive exploration of L2 and L3 features of SONiC, focusing specifically on its deployment, configuration, and behavioral testing using GNS3.

In this document, we will delve into the intricacies of deploying and configuring SONiC, showcasing its extensive capabilities in simulating complex network environments using GNS3. By leveraging GNS3, a powerful network simulation tool, we can create virtualized instances of SONiC, enabling us to thoroughly test and evaluate a variety of its features. Through practical demonstrations and step-by-step guidelines, we aim to equip readers with the necessary knowledge and insights to successfully implement SONiC in their own network environments.

Key areas of focus will include the deployment process, encompassing testbed setup, and initial configuration of SONiC. Furthermore, we will explore a range of L2 and L3 features within the GNS3 framework. By examining these features in a controlled, virtual environment, we can gain valuable insights into SONiC's capabilities, uncover any potential challenges, and identify best practices for maximizing its effectiveness.

As we embark on this journey into the world of SONiC and GNS3, we encourage readers to actively engage with the material, experiment with different configurations, and adapt the knowledge to suit their unique networking requirements. Together, we can unlock the true potential of SONiC and pave the way for a more open, scalable, and efficient future in network management.

Intended Audience

This document is tailored for network administrators, system integrators, and network engineers who are interested in exploring and implementing the Software for Open Networking in Cloud (SONiC) operating system within the context of GNS3. It is designed for individuals with a solid understanding of networking principles, including L2 and L3 protocols. Whether you are a hyperscaler, network operator, or vendor, this document aims to provide you with practical insights, step-by-step guidance, and best practices for deploying, configuring, and conducting behavioral testing of SONiC's L2 and L3 features using the GNS3 network simulation tool.

Tested Setup

To deploy any topology, we need a testbed that will set up the perfect environment where we can deploy our topologies. Now testbeds are of two types i.e., physical and virtual, depending on our availability of resources (switches, hosts, servers). If we have the required devices available for our topology, then we can go with a physical testbed otherwise we will opt for the virtual one.

SONiC Virtual Switch Testbed

To prepare the testbed, we need the following things:

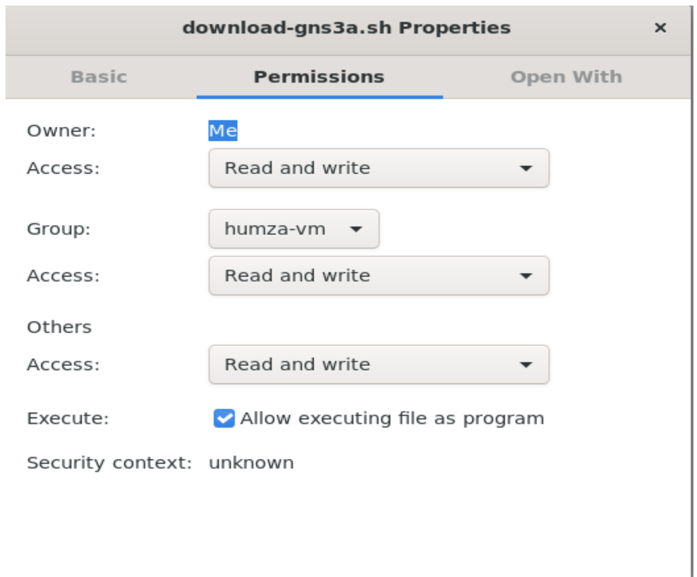
- GNS3
- Device image for GNS3
- SONiC image (.img file)
- The proper set of commands
- Only for SONiC versions>=201904

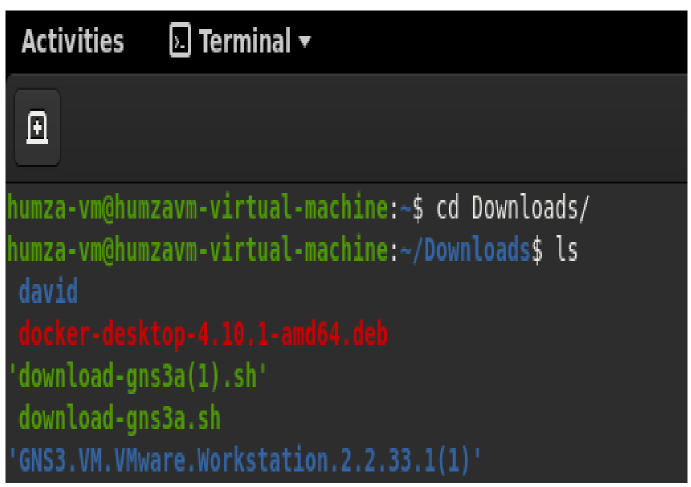
This document describes all steps one by one with necessary screenshots and links.

Device Image for GNS3

To deploy a SONiC image in GNS3, we need a device image. To download that image, the link is given here

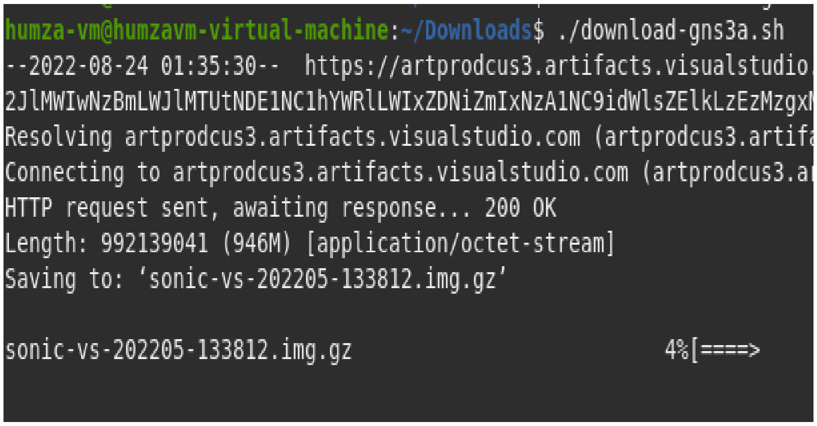
After downloading, some changings are required to make it compatible with GNS3. For Ubuntu, make it executable by following the path “right-click>properties>permissions>allow executing file as program”





After that make some changes to the file by using the command "sudo vi <filename>”. In our case, the command is “sudo vi download-gns3a.sh”. By using vim editor, press “i” to go to insert mode. SONiC image has different versions. In this testbed, version "202205" is used because it is the latest image. The figure, which is given below, explains where changes are made in the file.

- To run an executable file, write the following command "./download-gns3a.sh" in the terminal. Make sure that this command must be run in that directory in which "download-gns3a.sh" file is present.



After completing the above procedure, the device image, as well as the zip file of SONiC (.img.gz file), will be installed. To obtain a SONiC image (.img file), extract the zip file.

SONiC image for GNS3

To download SONiC image (.img file) for GNS3, the procedure is given below:

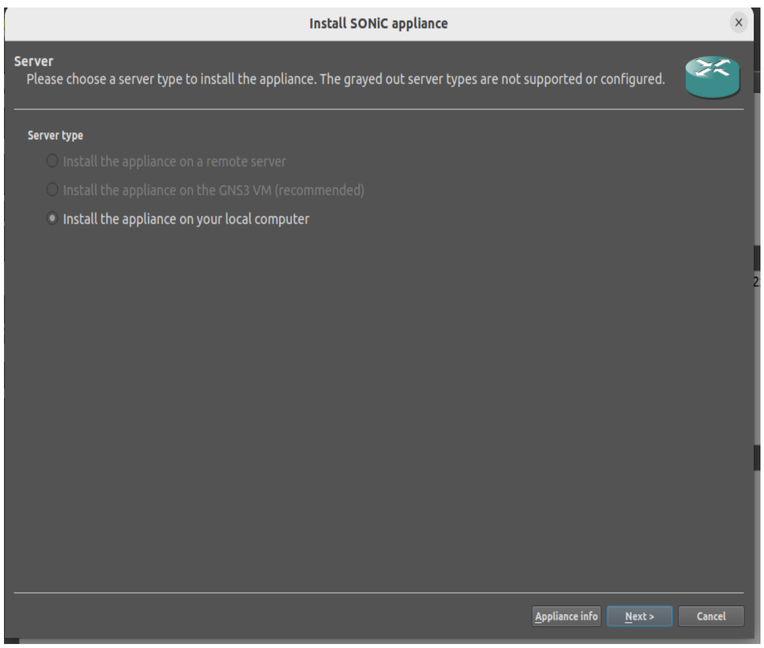
- By using the link which is given here, first zip file will be downloaded, and then extract the image file (.img)

There are many branches of SONiC at this site. In our case, "sonic-vs.img.gz" latest branch (202205) is used.

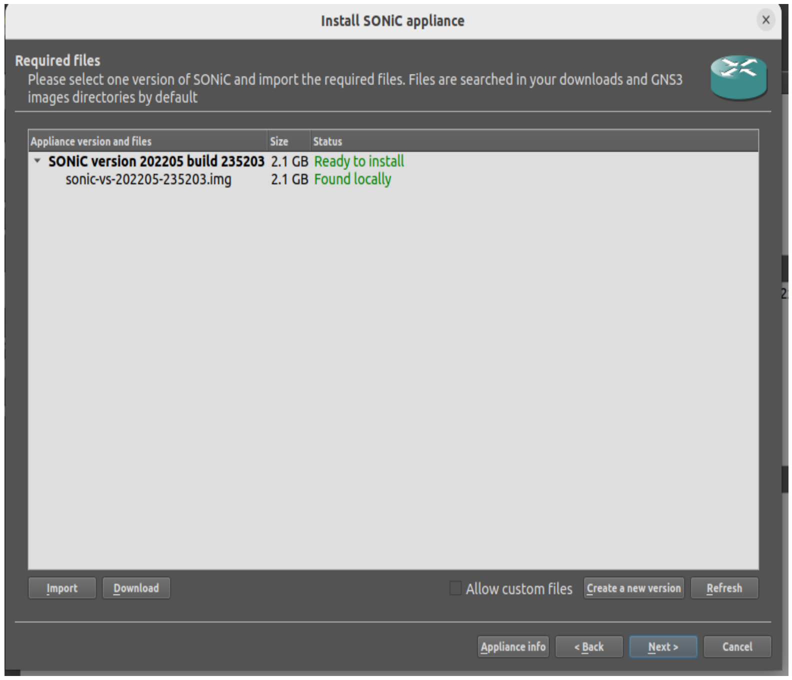
Importing device and SONiC image in GNS3

To import the device and SONiC image in GNS3 after creating a project, follow the path given below:

- file>import appliance
- After that, a pop-up menu is opened, Click on "Install the appliance on your local computer".



- Now click on the "Next" button. It shows a SONiC image version downloaded previously. Now import the SONiC image by clicking on the "import" button, choose the required image, and then click on the "Next" button.



Port Breakout

In SONiC, all ports are integral of 4 like Ethernet0, Ethernet4, Ethernet8, and so on. In GNS3, when connections are made with a switch, a pop-up menu is opened and shows which port one wants to use. If Ethernet1 is selected in GNS3, it means that Ethernet4 will be used in SONiC CLI. Ethernet2 in GNS3 is mapped with Ethernet8 in SONiC CLI.

Layer 2 Features

VLAN

Introduction

This section explains the step-by-step procedure to deploy VLAN topology and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Introduction to VLAN

A VLAN, or Virtual Local Area Network, is a logical grouping of devices on a network. In a traditional network, all devices are part of the same physical LAN, meaning that they are all on the same broadcast domain and can communicate with each other freely. However, with VLANs, a single physical network can be divided into multiple virtual networks, each with its own unique VLAN ID.

Without VLANs, a broadcast sent from host A would reach all devices on the network. Each device will receive and process broadcast frames, increasing the CPU overhead on each device and reducing the overall security of the network.

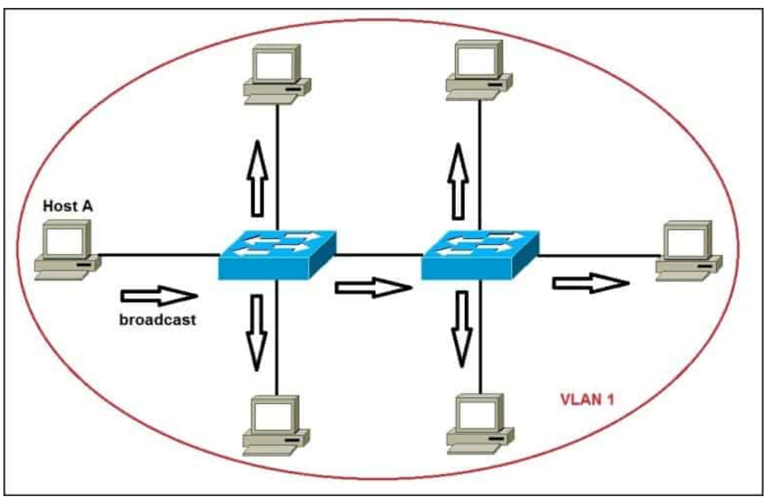


Fig: Topology with all hosts inside the same VLAN

Devices within the same VLAN can communicate with each other as if they were on the same physical LAN, but devices in different VLANs cannot communicate with each other unless specifically allowed by a router or switch. This can improve security by preventing unauthorized access to devices on the network, and can also improve network performance by reducing broadcast traffic.

By placing interfaces on both switches into a separate VLAN, a broadcast from host A would reach only devices inside the same VLAN, since each VLAN is a separate broadcast domain. Hosts in other VLANs will not even be aware that the communication took place. This is shown in the picture below:

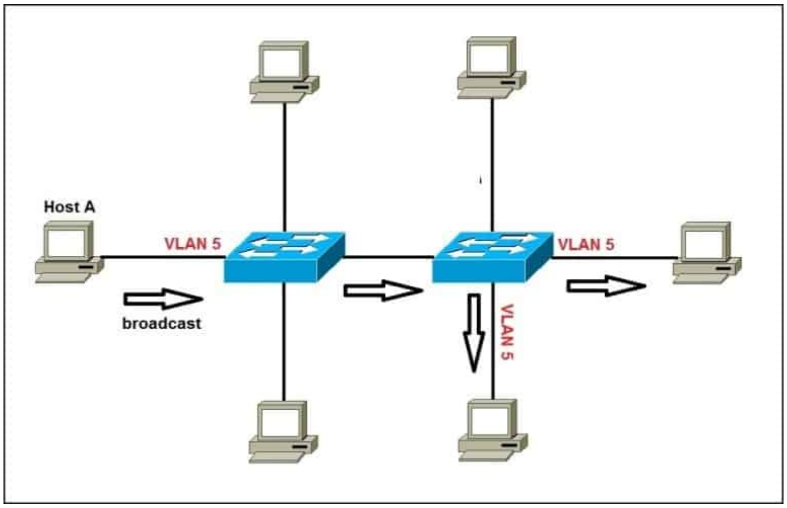


Fig: Topology hosts having different VLANs

NOTE: To reach hosts in a different VLAN, a router is needed.

Intra-Vlan

As the name suggests, “Intra” means “Inside”, Intra-VLAN communication refers to the ability of devices within the same VLAN to communicate with each other. Devices within the same VLAN are connected to the same broadcast domain, which means that they can communicate directly with each other without the need for routing.

To enable inter-VLAN communication, you will need to configure the devices with the appropriate IP addresses and subnet masks. The devices can then communicate with each other using these IP addresses.

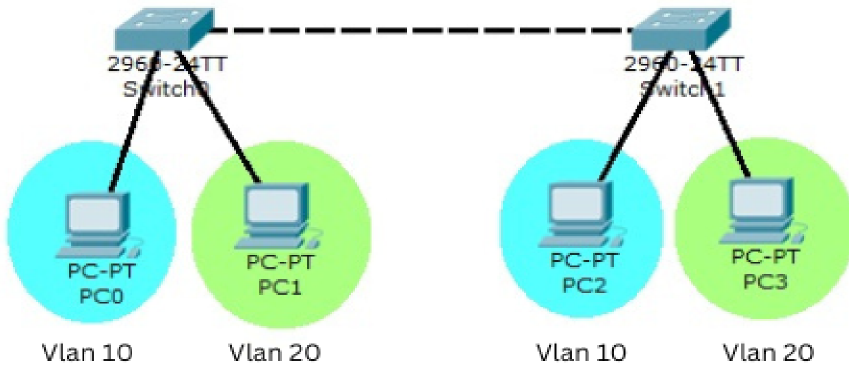


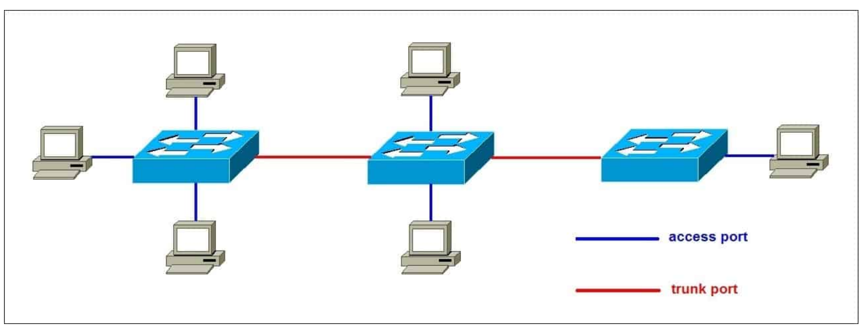
Fig: Intra-VLAN Topology using Multiple Switches

Trunk & Access Ports

If you intend to use VLANs in your network, you will need to configure some ports on a switch as access ports and trunk ports. Here is a description of each port type:

Access port – a port that can be assigned to a single VLAN. This type of interface is configured on switch ports that are connected to end devices such as workstations, printers, or access points.

Trunk port – a port that is connected to another switch. This type of interface can carry traffic from multiple VLANs, thus enabling you to extend VLANs across your entire network. Frames are tagged by assigning a VLAN ID to each frame as they traverse between switches.



Network Topology

After importing images, consider this simple example. Suppose we have a network with two departments: marketing, and finance. We want to create separate VLANs for each department to improve network security and performance.

To do this, we would need to configure our switches and hosts accordingly. Let's say we have two switches, S-1 and S-2, and four hosts, PC1 through PC4. Now draw network topology in GNS3 using SONiC switches and hosts.

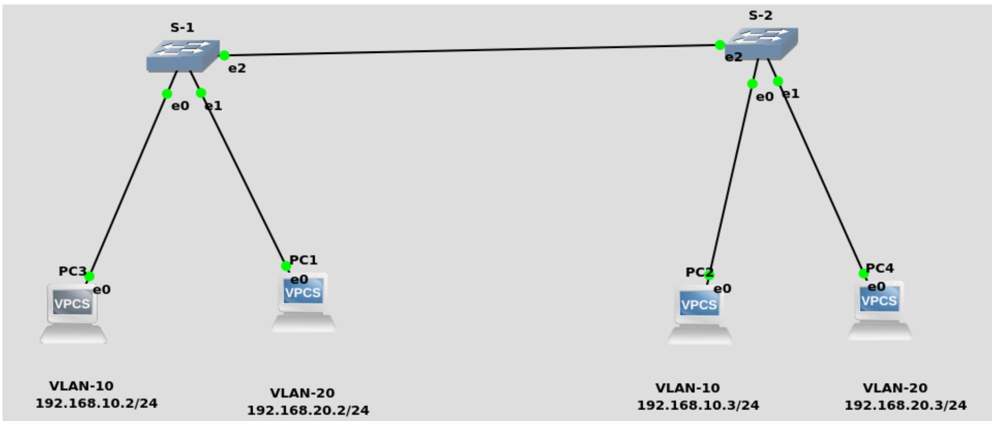


Fig: Network Topology

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured and the same steps are repeated for the switch (S-2). Command Reference guide is also available on GitHub for SONiC, whose link is given [here]

Follow these steps to configure Switch-1.

Step-1

- Check the status of interfaces by using the command:
  show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- Administrative ports (Admin) are used for device management and configuration, and allow administrators to remotely access and configure the device. Operational ports (Oper), on the other hand, are used for regular network traffic, such as passing data between devices on the network.
- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet64

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, the command is given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using the “sudo config reload” command.

By following the above commands, make changes in "config_db” and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them. Remove the IP to make that interface a switch port (L2). For this, commands are given below:

- sudo config interface ip remove/add <interface_name> <ip_addr>
- admin@sonic:~$ sudo config interface ip remove Ethernet64 10.11.12.13/31

IP must be removed from all those interfaces, which are to be used in network topology.

Note: It is better practice to save configurations after executing two or three commands.

Step-4

Now create VLANs for topology. Before creating VLANs, check the VLAN table by using the following command:

show vlan brief

```
admin@sonic:~$ show vlan brief
+-----------+-------------+--------+---------------+
| VLAN ID   | IP Address  | Ports  | Port Tagging  |
+===========+=============+========+===============+
+-----------+-------------+--------+---------------+
admin@sonic:~$
```

In the above table, no VLAN is created, so create VLANs by using the following command:

- sudo config vlan (add | del) <vlan_id>
- admin@sonic:~$ sudo config vlan add 10

```
---------+--------------+---------+---------------+
VLAN ID | IP Address   | Ports   | Port Tagging  |
========+==============+=========+===============+
     10 |              |         |               |
--------+--------------+---------+---------------+
     20 |              |         |               |
```

Step-5

Assign VLANs to interfaces. In SONiC, an interface can be tagged or un-tagged. Trunk ports should be tagged while access ports are un-tagged.

- sudo config vlan member add/del [-u|--untagged] <vlan_id> <member_portname>
- admin@sonic:~$ sudo config vlan member add 30 Ethernet8
  This command will add Ethernet8 as a member of vlan 30 and it is tagged.

- admin@sonic:~$ sudo config vlan member add -u 10 Ethernet0

This command will add Ethernet0 as a member of vlan 10 and it is un-tagged.

```
+-----------+--------------+-----------+----------------+-------------
 VLAN ID | IP Address    | Ports     | Port Tagging   | Proxy ARP
=========+===============+===========+================+=============
      10 |               | Ethernet0 | untagged       | disabled
         |               | Ethernet8 | tagged         |
---------+---------------+-----------+----------------+-------------
      20 |               | Ethernet4 | untagged       | disabled
         |               | Ethernet8 | tagged         |
```

Step-6

Repeat steps 1-5 for the switch (s-2).

Step-7

Assign IP addresses to hosts given in network topology.

```
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done

PC1> █
```

Note: It is highly recommended to save configurations in the host using the save command.

Result

After configuring the switches and hosts, hosts in the same VLAN can send traffic. In the figure below, it is clearly seen that PC1 can send traffic to PC4 because both are in the same VLAN i.e. VLAN20, while it cannot send traffic to PC3 due to a different VLAN. So, VLAN is successfully configured in the topology.

```
PC1> ping 192.168.20.3

84 bytes from 192.168.20.3 icmp_seq=1 ttl=64 time=9.538 ms
84 bytes from 192.168.20.3 icmp_seq=2 ttl=64 time=8.827 ms
84 bytes from 192.168.20.3 icmp_seq=3 ttl=64 time=9.021 ms
84 bytes from 192.168.20.3 icmp_seq=4 ttl=64 time=8.555 ms
84 bytes from 192.168.20.3 icmp_seq=5 ttl=64 time=8.631 ms

PC1> ping 192.168.10.3

host (255.255.255.0) not reachable

PC1>
```

Inter-VLAN

Introduction

This section explains the step-by-step procedure to deploy Inter-VLAN topology and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

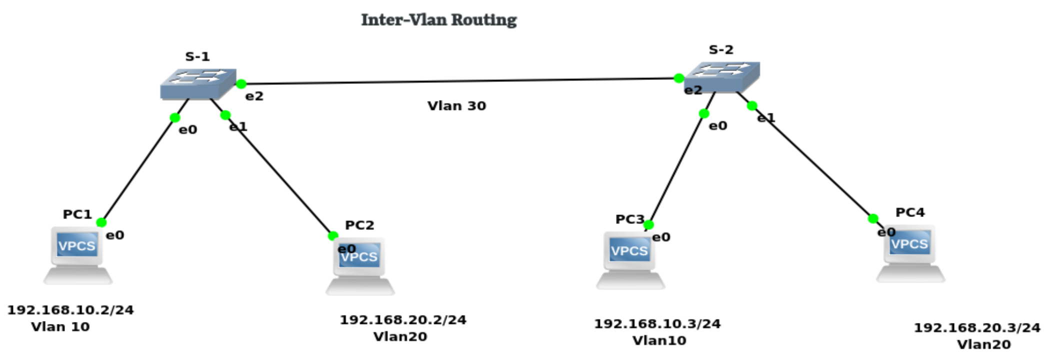


Fig: Network Topology

For the above topology, two switches (S-1, S-2) and four hosts are used.

Note: Be patient while using SONiC CLI because it takes some time to show results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured and the exact steps are repeated for the switch (S-2). The command reference guide is also available on GitHub for SONiC, whose link is given here

For the configuration of a switch (s-1), the steps are given below

Step-1

- Check the status of interfaces by using the command:
  show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A     fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A     fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A     fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A    fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A    fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A    fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A    fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A    fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A    fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A    fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A    fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A    fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A    fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A    fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A    fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A    fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A    fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure “config_db”, the command is given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using the “sudo config reload” command.

By following the above commands, make changes in "config_db" and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28"
        "alias": "fortyGigE0/0
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28"
        "alias": "fortyGigE0/0'
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32"
        "alias": "fortyGigE0/4'
-- INSERT --
```

Note: It is recommended to use the first method to change interface status. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them. Remove the IP to make that interface a switch port (L2). For this, commands are given below:

- sudo config interface ip remove/add <interface_name> <ip_addr>
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

IP must be removed from all those interfaces, which are to be used in network topology.

Note: It is better practice to save configurations after executing two or three commands.

Step-4

Now create VLANs for topology. Before creating VLANs, check VLAN table by using the following command:

show vlan brief

```
admin@sonic:~$ show vlan brief
+-----------+--------------+---------+---------------+
| VLAN ID   | IP Address   | Ports   | Port Tagging  |
+===========+==============+=========+===============+
+-----------+--------------+---------+---------------+
admin@sonic:~$
```

In the above table, no VLAN is created, so create VLANs by using the following command:

- sudo config vlan (add | del) <vlan_id>
- admin@sonic:~$ sudo config vlan add 10

```
--------+--------------+---------+---------------
VLAN ID | IP Address   | Ports   | Port Tagging
========+==============+=========+===============
     10 |              |         |
--------+--------------+---------+---------------
     20 |              |         |
```

Step-5

Assign VLANs to interfaces. In SONiC, an interface can be tagged or un-tagged. Trunk ports should be tagged while access ports are un-tagged.

- sudo config vlan member add/del [-u|--untagged] <vlan_id> <member_portname>
- admin@sonic:~$ sudo config vlan member add 30 Ethernet8

This command will add Ethernet8 as a member of vlan 30 and it is tagged.

- admin@sonic:~$ sudo config vlan member add -u 10 Ethernet0

This command will add Ethernet0 as a member of vlan 10 and it is un-tagged.

```
[[A+-----------+-------------+-----------+---------------+
|   VLAN ID | IP Address   | Ports     | Port Tagging  |
+===========+==============+===========+===============+
|        10 |              | Ethernet0 | untagged      |
|           |              | Ethernet8 | tagged        |
+-----------+--------------+-----------+---------------+
|        20 |              | Ethernet4 | untagged      |
|           |              | Ethernet8 | tagged        |
```

Step-6

Assign IP addresses to VLAN interfaces by using the following command:

- sudo config interface ip remove/add Vlan<vlan_id> <ip_addr
- admin@sonic:~$ sudo config interface ip add Vlan100 192.168.10.1/24

```
admin@sonic:~$ show vlan brief
+-----------+-----------------+-----------+----------------+
|   VLAN ID | IP Address      | Ports     | Port Tagging   |
+===========+=================+===========+================+
|        10 | 192.168.10.1/24 | Ethernet0 | untagged       |
|           |                 | Ethernet8 | tagged         |
+-----------+-----------------+-----------+----------------+
|        20 | 192.168.20.1/24 | Ethernet4 | untagged       |
|           |                 | Ethernet8 | tagged         |
+-----------+-----------------+-----------+----------------+
admin@sonic:~$
```

Note: It is highly recommended that while assigning ip address to VLAN, the name should be started with capital "V" like in our case "Vlan100". Otherwise, the command will not be executed.

Step-7

Repeat steps 1-6 for the switch (s-2).

Step-8

Assign IP addresses to hosts given in network topology.

```
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done

PC1>
```

Note: It is highly recommended to save configurations in the host using save command.

Result

After configuring the switches and hosts, hosts in the same as well as in different VLANs can send traffic. In the figure below, it is clearly seen that PC1 can send traffic to PC2, PC3, and PC4. So, Inter-VLAN is successfully implemented in the topology.

```
Activities    Terminal ▾

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

PC1> ping 192.168.20.3

84 bytes from 192.168.20.3 icmp_seq=1 ttl=63 time=16.571 ms
84 bytes from 192.168.20.3 icmp_seq=2 ttl=63 time=10.690 ms
84 bytes from 192.168.20.3 icmp_seq=3 ttl=63 time=9.623 ms
84 bytes from 192.168.20.3 icmp_seq=4 ttl=63 time=9.296 ms
84 bytes from 192.168.20.3 icmp_seq=5 ttl=63 time=9.154 ms

PC1> ping 192.168.20.2

84 bytes from 192.168.20.2 icmp_seq=1 ttl=63 time=5.806 ms
84 bytes from 192.168.20.2 icmp_seq=2 ttl=63 time=5.521 ms
84 bytes from 192.168.20.2 icmp_seq=3 ttl=63 time=5.360 ms
84 bytes from 192.168.20.2 icmp_seq=4 ttl=63 time=4.497 ms
84 bytes from 192.168.20.2 icmp_seq=5 ttl=63 time=4.740 ms

PC1> ping 192.168.10.3

84 bytes from 192.168.10.3 icmp_seq=1 ttl=64 time=9.402 ms
84 bytes from 192.168.10.3 icmp_seq=2 ttl=64 time=13.389 ms
84 bytes from 192.168.10.3 icmp_seq=3 ttl=64 time=11.738 ms
84 bytes from 192.168.10.3 icmp_seq=4 ttl=64 time=8.849 ms
84 bytes from 192.168.10.3 icmp_seq=5 ttl=64 time=12.225 ms

PC1>
```

Layer 3 (IPv4)

Static Routing

Introduction

This section explains the step-by-step procedure to test static routing and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

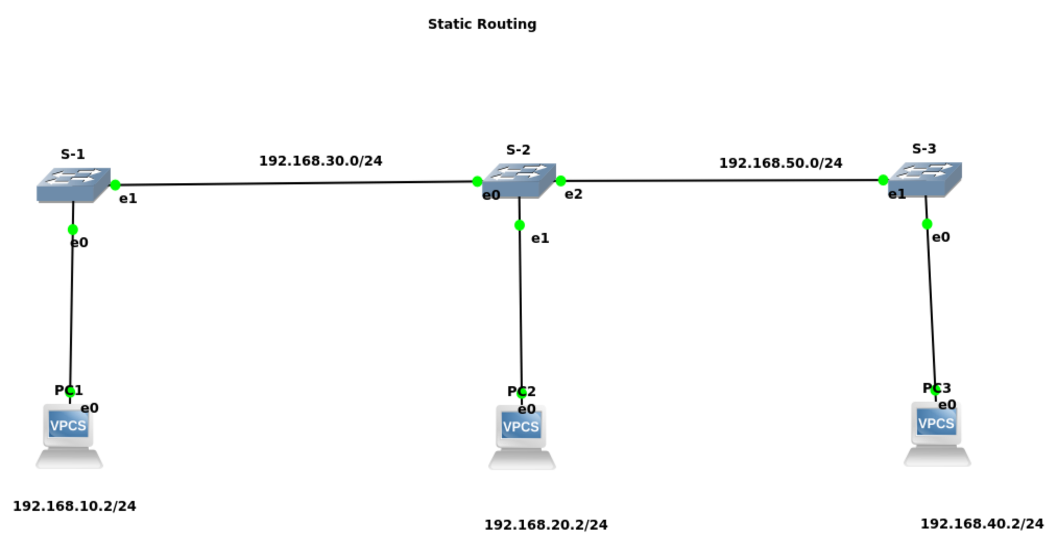


Fig: Network Topology

For the above topology, three switches (S-1, S-2 & S-3) and three hosts are used.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured and the exact steps are repeated for the switch (S-2). The command Reference guide is also available on GitHub for SONiC, whose link is given here

For the configuration of a switch (s-1), the steps are given below

Step-1

- Check the status of interfaces by using the command:
  show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, commands are given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using "sudo config reload" command.

By following the above commands, make changes in "config_db" and then save it.

Note: It is recommended to use the first method to change interface status. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them. Remove the IP to make that interface a switch port (L2). For this, commands are given below:

- sudo config interface ip remove/add <interface_name> <ip_addr>
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

IP must be removed from all those interfaces, which are to be used in network topology.

Note: It is better practice to save configurations after executing two or three commands.

Step-4

Assign gateway to s-1, s-2 and s-3 with 192.168.10.1/24, 192.168.20.1/24 and 192.168.40.1/24 respectively using the command given below:

- sudo config interface ip remove/add <interface_name> <ip_addr>

Step-5

In Static routing, all the existing static routes must be added in each router one by one by using network and "next-hop" addresses by using command given below:

- sudo config route add prefix [vrf <vrf>] <A.B.C.D/M> nexthop [vrf <vrf>] <A.B.C.D> dev <interface name>

This command is used to add a static route. Note that prefix/next-hop vrf's and interface name are optional.

- admin@sonic:~$ sudo config route add prefix 192.168.20.0/24 nexthop 192.168.30.3

```
35  sudo config route add prefix 192.168.20.0/24 nexthop 192.168.30.3
36  sudo config save -y
37  sudo config route add prefix 192.168.40.0/24 nexthop 192.168.30.3
38  sudo config save -y
39  show history
```

In above example, switch (s-1) is considered. IP Address after prefix, is the network address that is to be added in the router (s-1) and the address after next-hop is the interface address.

Step-6

Repeat steps 1-5 for the switch (S-2) and (S-3).

```
42  sudo config route add prefix 192.168.40.0/24 nexthop 192.168.50.2
43  sudo config save -y
```

```
 9  sudo config route add prefix 192.168.10.0/24 nexthop 192.168.50.3
10  sudo config save -y
11  show ip interfaces
12  sudo config save -y
13  sudo config route add prefix 192.168.20.0/24 nexthop 192.168.50.3
14  sudo config save -y
15  cd /etc/sonic/
```

Step-7

Assign IP addresses to hosts given in network topology.

```
PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done
```

Note: It is highly recommended to save configurations in the hosts using save command.

Result

After configuring the switches and hosts. In the figure below, it is clearly seen that host PC1 can send traffic to the other two networks and PC2 can also send traffic to other networks. So, Static Routing is successfully configured in the topology.

PC1

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

PC1> ping 192.168.20.2

84 bytes from 192.168.20.2 icmp_seq=1 ttl=62 time=12.056 ms
84 bytes from 192.168.20.2 icmp_seq=2 ttl=62 time=8.780 ms
84 bytes from 192.168.20.2 icmp_seq=3 ttl=62 time=14.097 ms
84 bytes from 192.168.20.2 icmp_seq=4 ttl=62 time=9.284 ms
84 bytes from 192.168.20.2 icmp_seq=5 ttl=62 time=9.203 ms

PC1> ping 192.168.40.2

84 bytes from 192.168.40.2 icmp_seq=1 ttl=61 time=19.921 ms
84 bytes from 192.168.40.2 icmp_seq=2 ttl=61 time=6.615 ms
84 bytes from 192.168.40.2 icmp_seq=3 ttl=61 time=6.686 ms
84 bytes from 192.168.40.2 icmp_seq=4 ttl=61 time=15.280 ms
84 bytes from 192.168.40.2 icmp_seq=5 ttl=61 time=13.360 ms
```

PC2

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

PC2> ping 192.168.10.2

84 bytes from 192.168.10.2 icmp_seq=1 ttl=62 time=14.583 ms
84 bytes from 192.168.10.2 icmp_seq=2 ttl=62 time=12.342 ms
84 bytes from 192.168.10.2 icmp_seq=3 ttl=62 time=9.002 ms
84 bytes from 192.168.10.2 icmp_seq=4 ttl=62 time=13.014 ms
84 bytes from 192.168.10.2 icmp_seq=5 ttl=62 time=9.430 ms

PC2> ping 192.168.40.2

84 bytes from 192.168.40.2 icmp_seq=1 ttl=62 time=9.240 ms
84 bytes from 192.168.40.2 icmp_seq=2 ttl=62 time=9.609 ms
84 bytes from 192.168.40.2 icmp_seq=3 ttl=62 time=9.702 ms
84 bytes from 192.168.40.2 icmp_seq=4 ttl=62 time=8.875 ms
84 bytes from 192.168.40.2 icmp_seq=5 ttl=62 time=10.264 ms
```

RIP

Introduction

This section explains the step-by-step procedure to test RIP (routing protocol) and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to [Testbed](#) section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

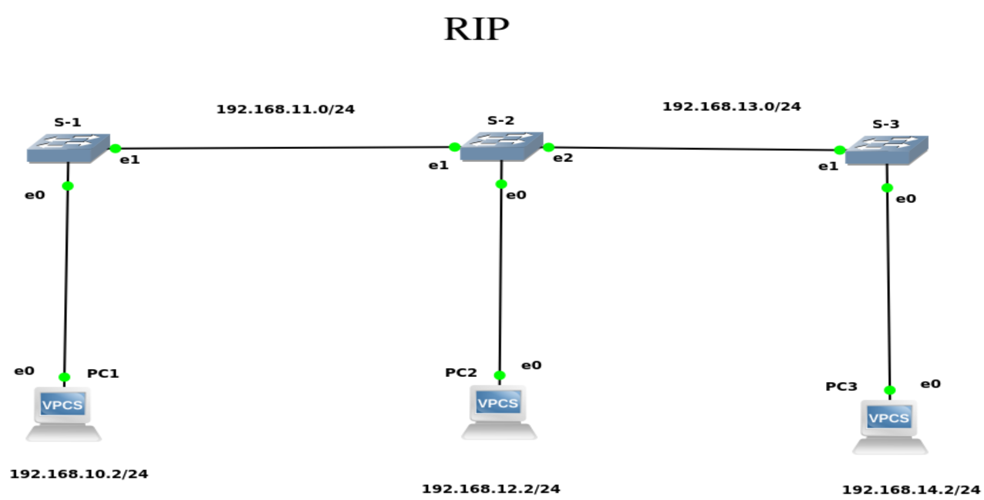


Fig: Network Topology

For the above topology, three switches (S-1, S-2 & S-3) and three hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

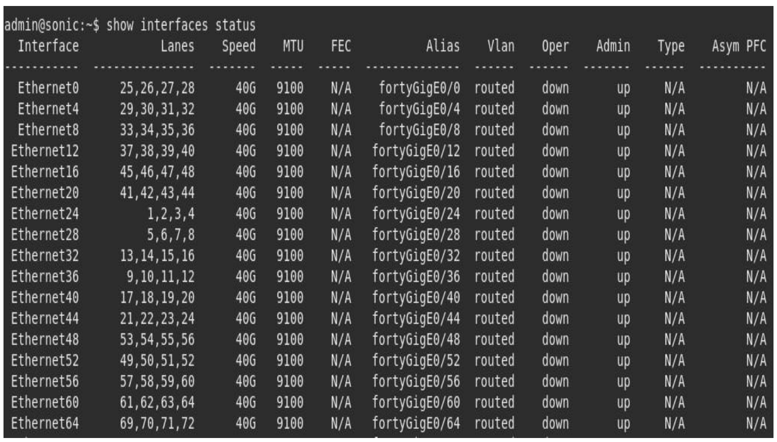
For the above topology, all hosts and switches are first configured before sending traffic. First, switch (S-1) is configured, and the same steps are repeated for the switch (S-2). Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status



```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using "sudo config reload" command.

By following the above commands, make changes in "config_db" and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
-- INSERT --
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces

```
admin@sonic:~$ show ip interfaces
Interface    Master    IPv4 address/mask    Admin/Oper
-----------  --------  -------------------  ----------
Ethernet0              10.0.0.0/31          up/up
Ethernet4              10.0.0.2/31          up/up
Ethernet8              10.0.0.4/31          up/up
```

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

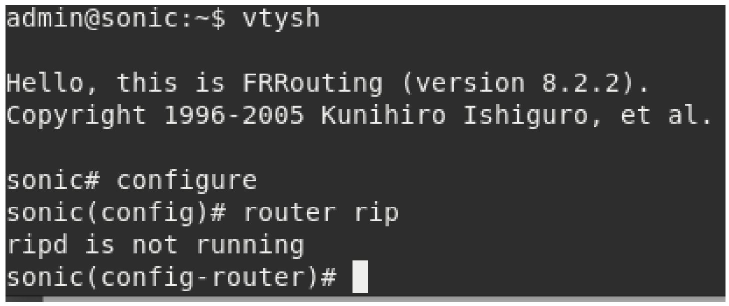
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, RIP daemon is not running. For ripd to be in the running mode use the following commands given below:

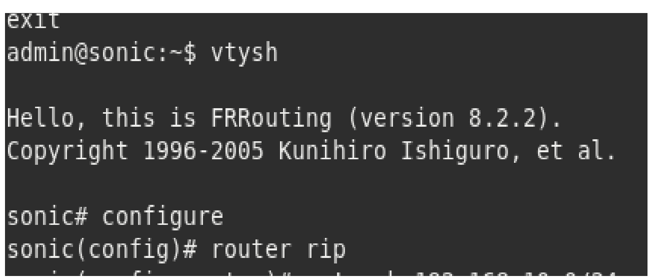
docker exec -it bgp bash
cd usr/lib/frr
ls
./ripd &



Step-5

RIP must be enabled before carrying out any of the RIP commands by using the command given below:

vtysh
configure
router rip



Step-6

After enabling RIP, assign all the network addresses to which the switch is directly connected by using the command given below:

network <network address>

network <interface name>

```
sonic# configure
sonic(config)# router rip
sonic(config-router)# network 192.168.10.0/24
sonic(config-router)# network Ethernet0
sonic(config-router)# network 192.168.11.0/24
sonic(config-router)# network Ethernet4
sonic(config-router)# exit
sonic(config)# exit
```

```
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ripd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
sonic#
```

Note: It is highly recommended to save the configurations using the "write" command.

Step-7

Repeat steps 1-6 for other switches as well.

Step-8

Check RIP status using the command given below:

show ip rip status

```
sonic# show ip rip status
Routing Protocol is "rip"
  Sending updates every 30 seconds with +/-50%, next due in 4 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
  Redistributing:
  Default version control: send version 2, receive any version
    Interface        Send  Recv   Key-chain
    Ethernet0        2     1 2
    Ethernet4        2     1 2
  Routing for Networks:
    192.168.10.0/24
    192.168.11.0/24
    Ethernet4
    Ethernet0
  Routing Information Sources:
    Gateway          BadPackets BadRoutes  Distance Last Update
    192.168.12.2             0          0       120   00:00:13
  Distance: (default is 120)
sonic#
```

Step-9

Assign IP addresses to hosts given in network topology.

```
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done

PC1>
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

After configuring the switches and hosts, hosts can send traffic. In the figure below, it is clearly seen that host PC1 can send traffic to PC2 and PC3. So, RIP is successfully configured in the topology.

```
PC1
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

PC1> ping 192.168.14.2

84 bytes from 192.168.14.2 icmp_seq=1 ttl=61 time=32.605 ms
84 bytes from 192.168.14.2 icmp_seq=2 ttl=61 time=18.026 ms
84 bytes from 192.168.14.2 icmp_seq=3 ttl=61 time=13.764 ms
84 bytes from 192.168.14.2 icmp_seq=4 ttl=61 time=14.593 ms
84 bytes from 192.168.14.2 icmp_seq=5 ttl=61 time=13.024 ms

PC1> ping 192.168.12.2

84 bytes from 192.168.12.2 icmp_seq=1 ttl=62 time=21.241 ms
84 bytes from 192.168.12.2 icmp_seq=2 ttl=62 time=9.829 ms
84 bytes from 192.168.12.2 icmp_seq=3 ttl=62 time=10.841 ms
84 bytes from 192.168.12.2 icmp_seq=4 ttl=62 time=9.619 ms
84 bytes from 192.168.12.2 icmp_seq=5 ttl=62 time=10.015 ms
```

EIGRP

Introduction

This section explains the step-by-step procedure to test EIGRP (routing protocol) and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

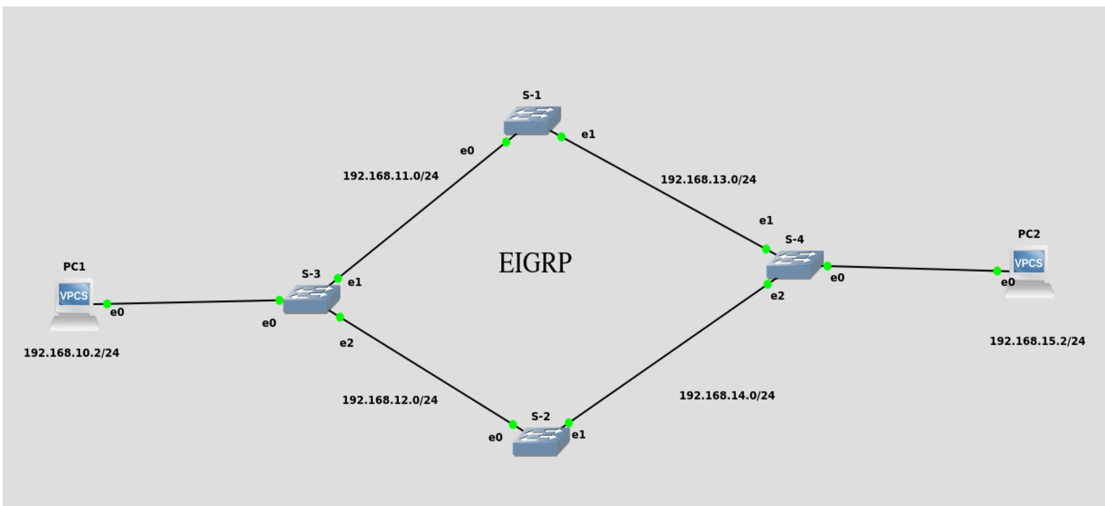


Fig: Network Topology

For the above topology, four switches (S-1, S-2, S-3 & S-4) and two hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

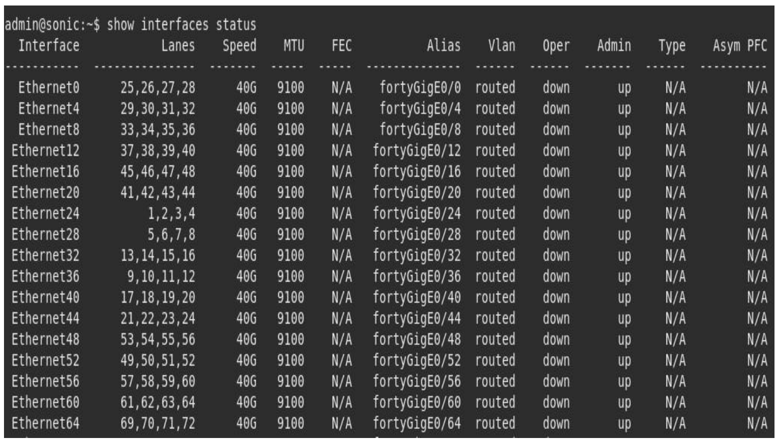
For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured and the exact steps are repeated for the other switches. A command reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-3.

Step-1

Check the status of interfaces by using the command:

- show interfaces status



```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A     fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A     fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A     fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A    fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A    fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A    fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A    fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A    fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A    fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A    fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A    fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A    fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A    fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A    fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A    fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A    fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A    fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using “sudo config reload” command.

By following the above commands, make changes in "config_db” and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

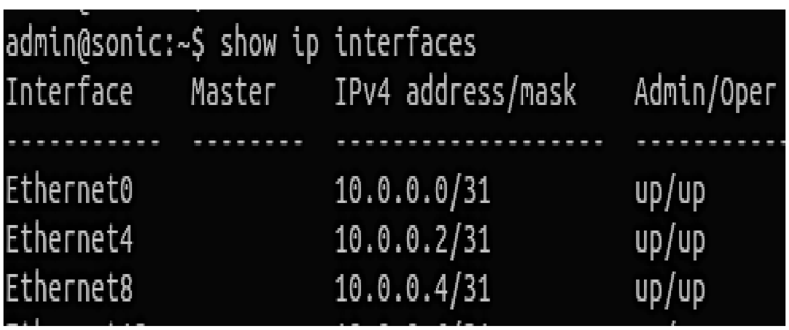
```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
-- INSERT --
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



```
admin@sonic:~$ show ip interfaces
Interface    Master    IPv4 address/mask    Admin/Oper
-----------  --------  -------------------  ----------
Ethernet0              10.0.0.0/31          up/up
Ethernet4              10.0.0.2/31          up/up
Ethernet8              10.0.0.4/31          up/up
```

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, EIGRP daemon is not running. For eigrpd to be in the running mode, use the following commands given below:

docker exec -it bgp bash
cd usr/lib/frr
ls
./eigrpd &

```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd usr/lib/frr
root@sonic:/usr/lib/frr# ls
babeld  eigrpd     frrcommon.sh  ldpd    ospfd  pimd    staticd   watchfrr.sh
bfdd    fabricd    frrinit.sh    nhrpd   pathd  ripd    vrrpd     zebra
bgpd    frr-reload isisd         ospf6d  pbrd   ripngd  watchfrr
root@sonic:/usr/lib/frr# ./eigrpd &
[1] 92
root@sonic:/usr/lib/frr#
```

Step-5

EIGRP must be enabled before carrying out any of the EIGRP commands by using the command given below:

vtysh
configure
router eigrp <process-id> [vrf NAME]

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router eigrp 1
sonic(config-router)#
```

Specify vrf NAME if you want eigrp to work within the specified vrf. The process-ID in EIGRP is a numeric value between 1 and 65535.

Step-6

After enabling EIGRP, assign all the network addresses to which the switch is directly connected by using the command given below:

network <network address>

```
sonic# configure
sonic(config)# router eigrp 1
sonic(config-router)# network 192.168.10.0/24
sonic(config-router)# network 192.168.11.0/24
sonic(config-router)# network 192.168.12.0/24
sonic(config-router)#
```

```
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/eigrpd.conf
Configuration saved to /etc/frr/staticd.conf
sonic#
```

Note: It is highly recommended to save the configurations using the "write" command.

Step-7

Repeat steps 1-6 for other switches as well.

Step-8

Check EIGRP status using the command given below:

show ip eigrp topology

```
S-1
onnected to localhost.
scape character is '^]'.

onic# show ip eigrp topology

IGRP Topology Table for AS(1)/ID(192.168.13.2)

odes: P - Passive, A - Active, U - Update, Q - Query, R - Reply
       r - reply Status, s - sia Status

   192.168.10.0/24, 1 successors, FD is 30720, serno: 0
        via 192.168.11.2 (30720/28160), Ethernet0
   192.168.11.0/24, 1 successors, FD is 28160, serno: 0
        via Connected, Ethernet0
   192.168.12.0/24, 1 successors, FD is 30720, serno: 0
        via 192.168.11.2 (30720/28160), Ethernet0
   192.168.13.0/24, 1 successors, FD is 28160, serno: 0
        via Connected, Ethernet4
   192.168.14.0/24, 1 successors, FD is 30720, serno: 0
        via 192.168.13.3 (30720/28160), Ethernet4
   192.168.15.0/24, 1 successors, FD is 30720, serno: 0
        via 192.168.13.3 (30720/28160), Ethernet4
onic#
```

Step-9

Assign IP addresses to hosts given in network topology.

```
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done

PC1>
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

After configuring the switches and hosts, hosts can send traffic. In the figure below, it is clearly seen that host PC1 can send traffic to all the other networks and PC2. So, EIGRP is successfully configured in the topology.

```
PC1
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

PC1> ping 192.168.15.2

84 bytes from 192.168.15.2 icmp_seq=1 ttl=61 time=51.159 ms
84 bytes from 192.168.15.2 icmp_seq=2 ttl=61 time=14.099 ms
84 bytes from 192.168.15.2 icmp_seq=3 ttl=61 time=12.879 ms
84 bytes from 192.168.15.2 icmp_seq=4 ttl=61 time=14.753 ms
84 bytes from 192.168.15.2 icmp_seq=5 ttl=61 time=13.952 ms

PC1> ping 192.168.14.3

84 bytes from 192.168.14.3 icmp_seq=1 ttl=62 time=12.082 ms
84 bytes from 192.168.14.3 icmp_seq=2 ttl=62 time=12.507 ms
84 bytes from 192.168.14.3 icmp_seq=3 ttl=62 time=11.152 ms
84 bytes from 192.168.14.3 icmp_seq=4 ttl=62 time=11.546 ms
84 bytes from 192.168.14.3 icmp_seq=5 ttl=62 time=15.335 ms

PC1>
```

OSPF

Introduction

This section explains the step-by-step procedure to test OSPF (routing protocol) and configure features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

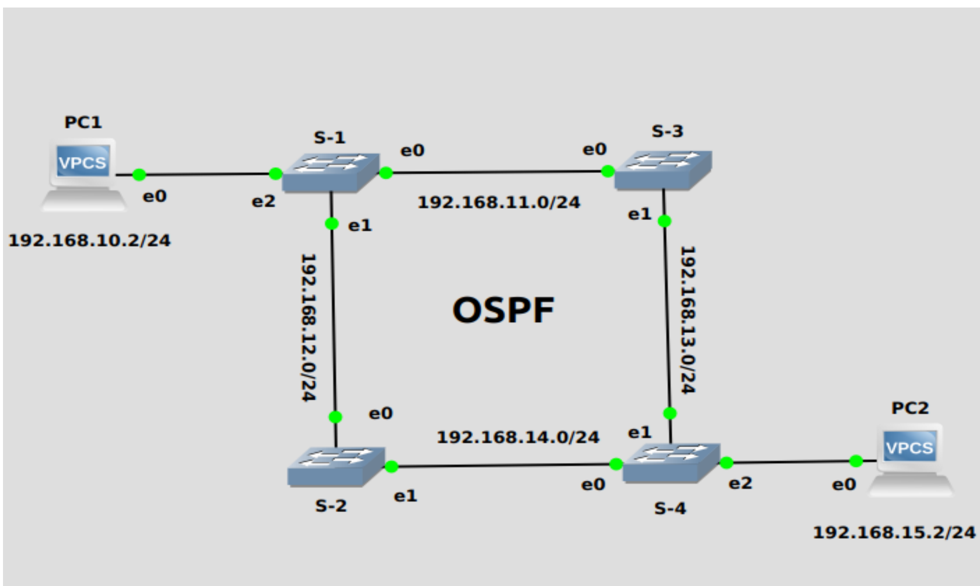


Fig: Network Topology

For the above topology, four switches (S-1, S-2, S-3 & S-4) and two hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that, first save all the configurations and then reload it using "sudo config reload" command.

By following the above commands, make changes in "config_db" and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
-- INSERT --
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1^{st} command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces

```
admin@sonic:~$ show ip interfaces
Interface    Master    IPv4 address/mask    Admin/Oper
-----------  --------  -------------------  ----------
Ethernet0              10.0.0.0/31          up/up
Ethernet4              10.0.0.2/31          up/up
Ethernet8              10.0.0.4/31          up/up
```

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

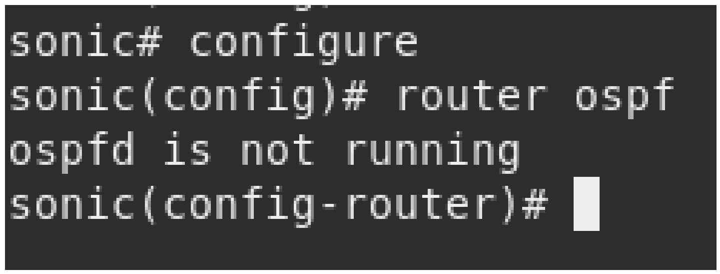
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, OSPF daemon is not running. For ospfd to be in the running mode use the following commands given below:

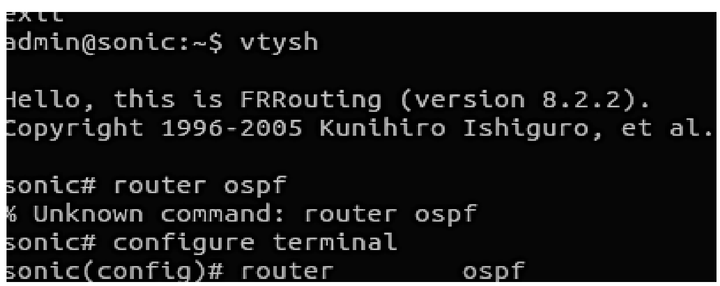
docker exec -it bgp bash
cd usr/lib/frr
ls
./ospfd &



Step-5

OSPF must be enabled before carrying out any of the ospf commands by using the commands given below:

vtysh
configure
router ospf



Step-6

After enabling OSPF, assign all the network addresses to which the switch is directly connected by using the command given below:

network <network address> area <address>

```
sonic(config)# router      ospf
sonic(config-router)# router  ospf
sonic(config-router)# network 192.168.10.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.11.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.12.0/24 area 0.0.0.0
sonic(config-router)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-7

Repeat steps 1-6 for other switches as well.

Step-8

Check ospf status using the commands given below:

show ip ospf

show ip ospf database router <router id>

```
sonic# show ip ospf
 OSPF Routing Process, Router ID: 192.168.12.2
 Supports only single TOS (TOS0) routes
 This implementation conforms to RFC2328
 RFC1583Compatibility flag is disabled
 OpaqueCapability flag is disabled
 Initial SPF scheduling delay 0 millisec(s)
 Minimum hold time between consecutive SPFs 50 millisec(s)
 Maximum hold time between consecutive SPFs 5000 millisec(s)
 Hold time multiplier is currently 1
 SPF algorithm last executed 27m52s ago
 Last SPF duration 518 usecs
 SPF timer is inactive
 LSA minimum interval 5000 msecs
 LSA minimum arrival 1000 msecs
 Write Multiplier set to 20
 Refresh timer 10 secs
 Maximum multiple paths(ECMP) supported  256
 Number of external LSA 0. Checksum Sum 0x00000000
 Number of opaque AS LSA 0. Checksum Sum 0x00000000
 Number of areas attached to this router: 1
 Area ID: 0.0.0.0 (Backbone)
```

```
sonic# show ip ospf database router 192.168.12.2

       OSPF Router with ID (192.168.12.2)

                Router Link States (Area 0.0.0.0)

  LS age: 54
  Options: 0x2  : *|-|-|-|-|-|E|-
  LS Flags: 0x3
  Flags: 0x0
  LS Type: router-LSA
  Link State ID: 192.168.12.2
  Advertising Router: 192.168.12.2
  LS Seq Number: 8000000c
  Checksum: 0x87be
  Length: 60

   Number of Links: 3

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.11.2
     (Link Data) Router Interface address: 192.168.11.2
      Number of TOS metrics: 0
       TOS 0 Metric: 10000

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.12.2
```

Step-9

Assign IP addresses to hosts given in network topology.

```
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done
```

Result

After configuring the switches and hosts, hosts can send traffic. In the figure below, it is clearly seen that host PC1 can send traffic to PC2 and other networks. So, OSPF is successfully configured in the topology.

```
PC1> ping 192.168.15.2

84 bytes from 192.168.15.2 icmp_seq=1 ttl=61 time=21.494 ms
84 bytes from 192.168.15.2 icmp_seq=2 ttl=61 time=12.209 ms
84 bytes from 192.168.15.2 icmp_seq=3 ttl=61 time=12.787 ms
84 bytes from 192.168.15.2 icmp_seq=4 ttl=61 time=14.255 ms
84 bytes from 192.168.15.2 icmp_seq=5 ttl=61 time=13.045 ms

PC1> ping 192.168.14.2

84 bytes from 192.168.14.2 icmp_seq=1 ttl=63 time=8.232 ms
84 bytes from 192.168.14.2 icmp_seq=2 ttl=63 time=7.322 ms
84 bytes from 192.168.14.2 icmp_seq=3 ttl=63 time=6.396 ms
84 bytes from 192.168.14.2 icmp_seq=4 ttl=63 time=6.017 ms
84 bytes from 192.168.14.2 icmp_seq=5 ttl=63 time=6.929 ms

PC1> ping 192.168.11.2

84 bytes from 192.168.11.2 icmp_seq=1 ttl=64 time=2.646 ms
84 bytes from 192.168.11.2 icmp_seq=2 ttl=64 time=2.757 ms
84 bytes from 192.168.11.2 icmp_seq=3 ttl=64 time=2.719 ms
84 bytes from 192.168.11.2 icmp_seq=4 ttl=64 time=3.137 ms
84 bytes from 192.168.11.2 icmp_seq=5 ttl=64 time=2.218 ms
```

Testing OSPF Link Cost in SONiC

Introduction

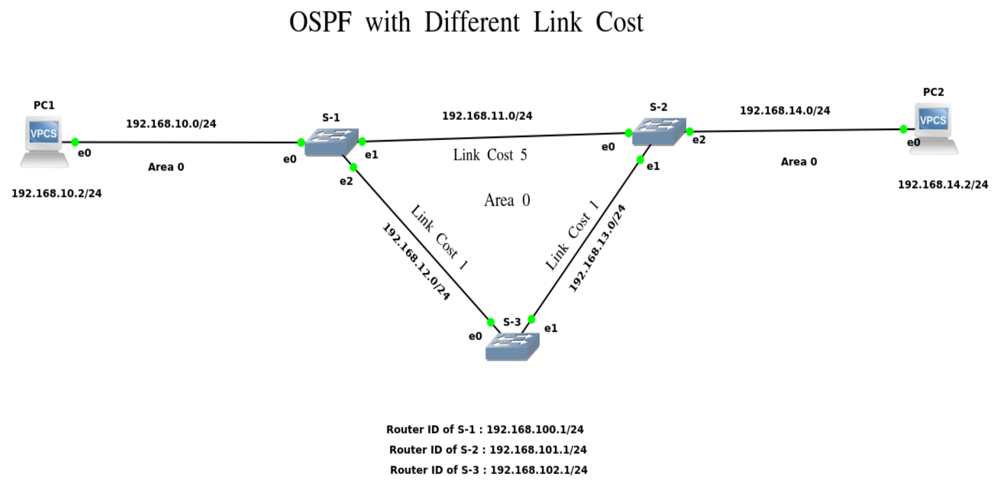
This section explains the step-by-step procedure to test OSPF Link Cost and configure its features by running related commands in SONiC CLI.

Testbed Setup

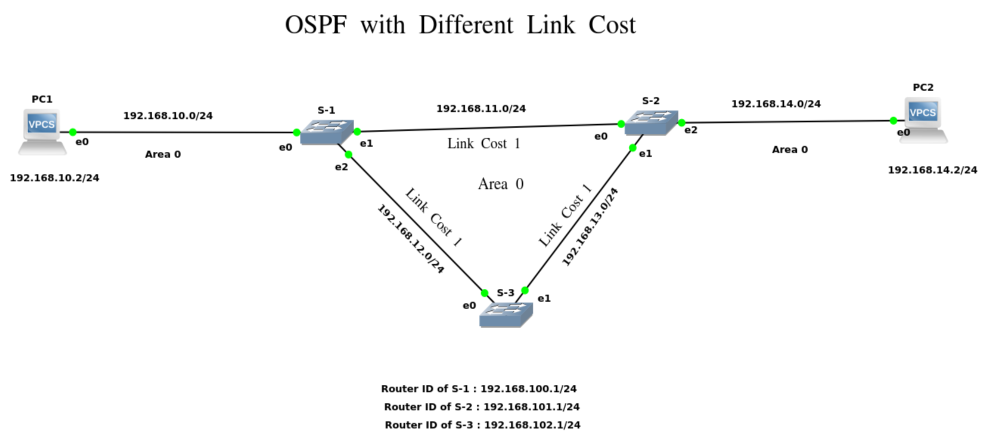
Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.



Fig#1: Network Topology



Fig#2: Network Topology

For the above topology, three switches (S-1, S-2 & S-3) and two hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the exact steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring "config_db" and the path is /etc/sonic. To configure "config_db", the command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using "sudo config reload" command.

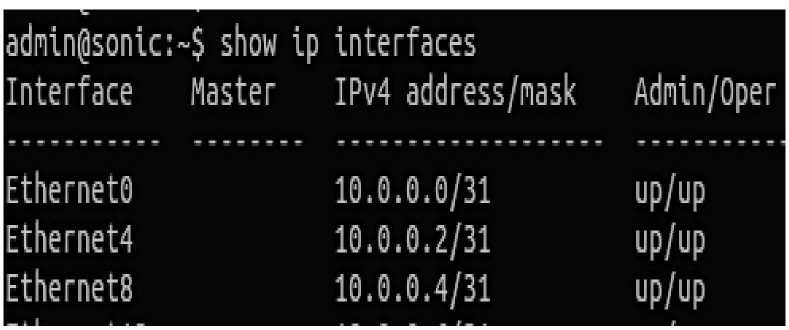
By following the above commands, make changes in "config_db" and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



admin@sonic:~$ show ip interfaces

Interface	Master	IPv4 address/mask	Admin/Oper
Ethernet0		10.0.0.0/31	up/up
Ethernet4		10.0.0.2/31	up/up
Ethernet8		10.0.0.4/31	up/up

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

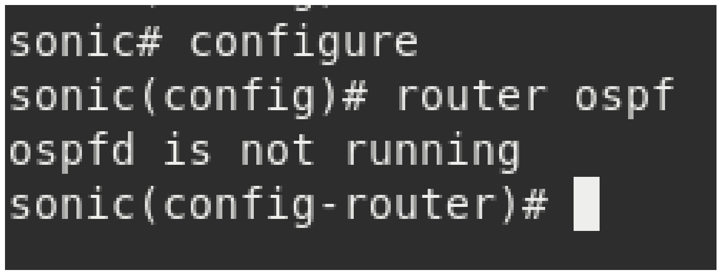
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

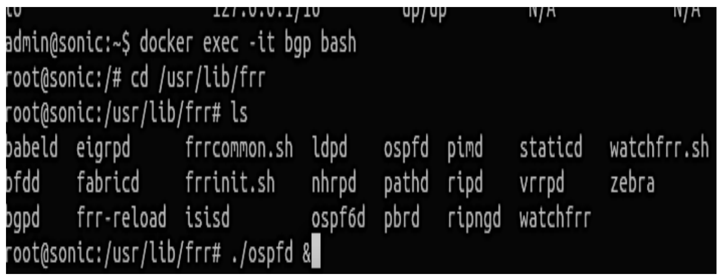
Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, the OSPF daemon is not running. For ospfd to be in the running mode use the following commands given below:

docker exec -it bgp bash
cd usr/lib/frr
ls
./ospfd &

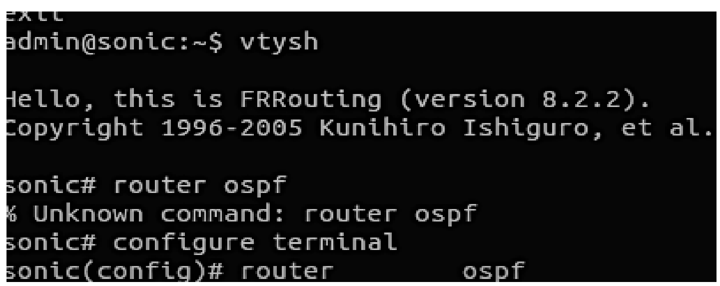




Step-5

OSPF must be enabled before carrying out any of the OSPF commands by using the commands given below:

vtysh
configure
router ospf



Step-6

OSPF can also be enabled on a per-interface basis. In this guide, all the configurations are made on a per-interface basis. To enable OSPF, use the following commands given below:

interface <interface name>

ip ospf area AREA

```
sonic# configure
sonic(config)# router ospf
sonic(config-router)# ospf router-id 192.168.102.1
sonic(config-router)# interface Ethernet0
sonic(config-if)# ip ospf area 0
```

Note: It is better practice to change Router-ID before enabling OSPF on a per-interface basis by using command “ospf router-id <A.B.C.D>”. Notice that, mixing both network commands (network) and interface commands (ip ospf) on the same router is not supported. If (ip ospf) is present, (network) commands will fail.

Step-7

After enabling OSPF, change link cost to a specific value by using the command given below:

ip ospf cost (1-65535)

The above command sets link cost for the specified interface. The cost value is set to router-LSA’s metric field and used for SPF calculation.

```
sonic(config)# interface Ethernet4
sonic(config-if)# ip ospf area 0
sonic(config-if)# ip ospf cost 5
```

```
sonic(config-if)# exit
sonic(config)# interface Ethernet8
sonic(config-if)# ip ospf area 0
sonic(config-if)# ip ospf cost 1
sonic(config-if)# exit
sonic(config)# exit
sonic# write
```

Step-8

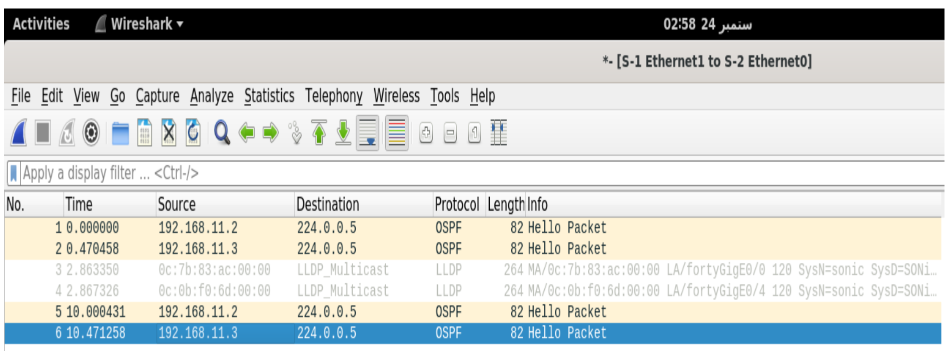
To check the status of link cost, use the following command given below:

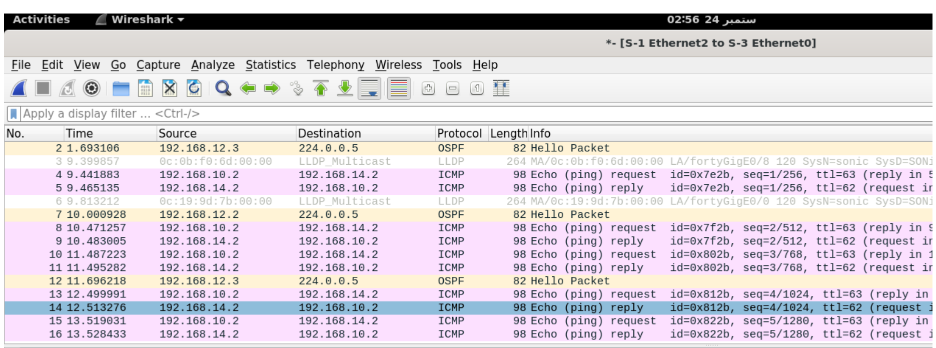
show ip ospf interface <interface name>

```
sonic# show ip ospf interface Ethernet4
Ethernet4 is up
  ifindex 20, MTU 9100 bytes, BW 10 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.11.2/24, Broadcast 192.168.11.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 192.168.100.1, Network Type BROADCAST, Cost: 5
  Transmit Delay is 1 sec, State Backup, Priority 5
  Designated Router (ID) 192.168.101.1 Interface Address 192.168.11.3/24
  Backup Designated Router (ID) 192.168.100.1, Interface Address 192.168.11.2
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.725s
```

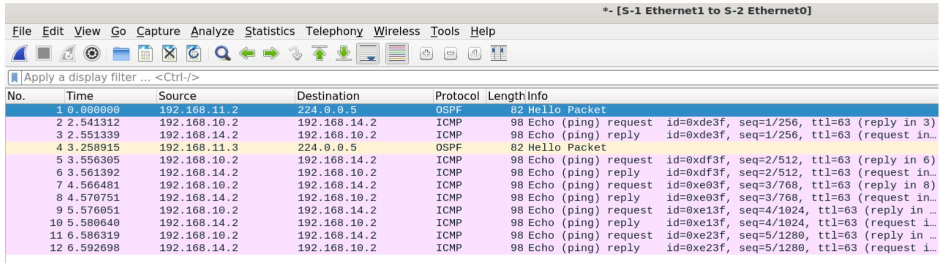
Step-9

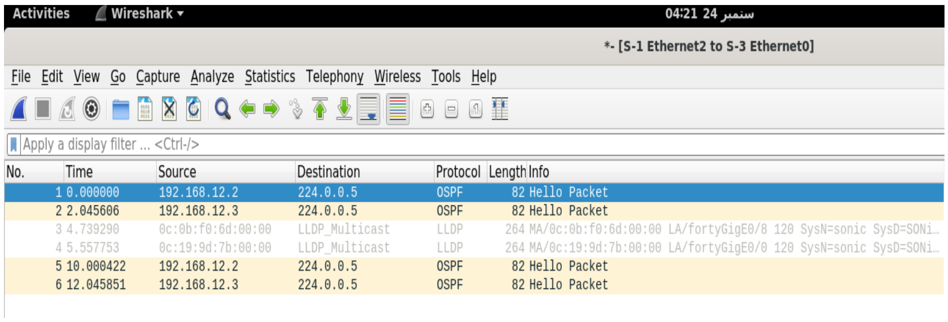
By using Wireshark, it is also verified that packets pass through that link whose cost is low. In Fig#1, from S-1 to S-2, the link cost is 5 while S-1 to S-3 and S-3 to S-2, the link cost is 2. So, packets do not pass-through S-1 to S-2 rather they pass through S-1 to S-3 and then S-3 to S-2.





In Fig#2, the link cost from S-1 to S-2 has been changed to 1 and the total cost from S-1 to S-3 and S-3 to S-2 has been changed to 2, so packets pass-through S-1 to S-2 due to low link cost.





Step-10

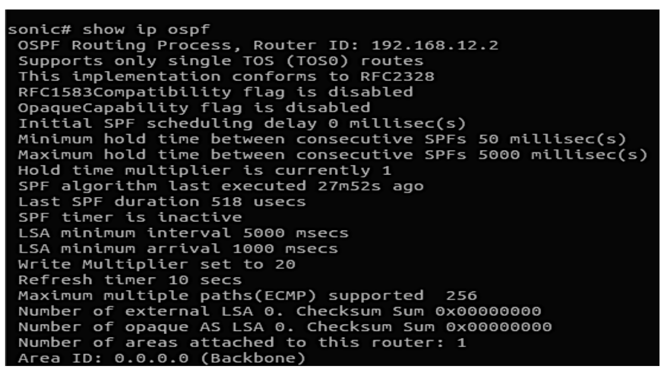
Repeat steps 1-9 for other switches as well.

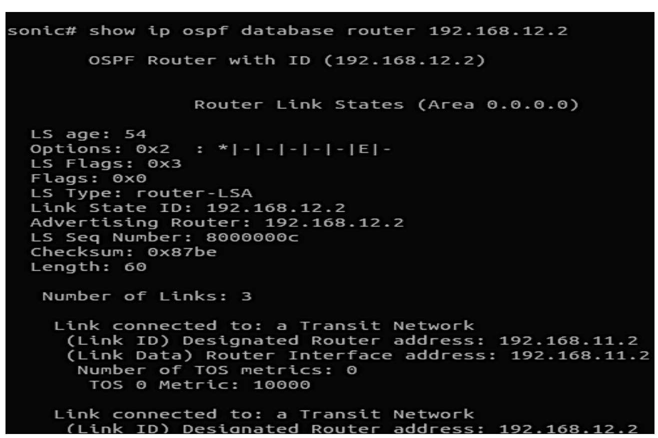
Step-11

Check OSPF status using the commands given below:

show ip ospf

show ip ospf database router <router id>





Step-12

Assign IP addresses to hosts given in network topology.

```
PC2> ip 192.168.19.2 192.168.19.1
Checking for duplicate address...
PC2 : 192.168.19.2 255.255.255.0 gateway 192.168.19.1

PC2> save
Saving startup configuration to startup.vpc
.  done

PC2>
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

After configuring the switches and hosts, hosts can send traffic. In the figure below, it is clearly seen that host PC1 can send traffic to PC2 and other networks. It is also verified that traffic pass-through that link cost is low. So, OSPF with different link costs is successfully configured in the topology.

```
PC1> ping 192.168.14.2

84 bytes from 192.168.14.2 icmp_seq=1 ttl=61 time=28.791 ms
84 bytes from 192.168.14.2 icmp_seq=2 ttl=61 time=15.871 ms
84 bytes from 192.168.14.2 icmp_seq=3 ttl=61 time=12.007 ms
84 bytes from 192.168.14.2 icmp_seq=4 ttl=61 time=18.786 ms
84 bytes from 192.168.14.2 icmp_seq=5 ttl=61 time=14.648 ms

PC1> ping 192.168.13.3

84 bytes from 192.168.13.3 icmp_seq=1 ttl=63 time=3.978 ms
84 bytes from 192.168.13.3 icmp_seq=2 ttl=63 time=6.867 ms
84 bytes from 192.168.13.3 icmp_seq=3 ttl=63 time=7.417 ms
84 bytes from 192.168.13.3 icmp_seq=4 ttl=63 time=8.354 ms
84 bytes from 192.168.13.3 icmp_seq=5 ttl=63 time=6.767 ms
```

OSPF Passive Interface

Introduction

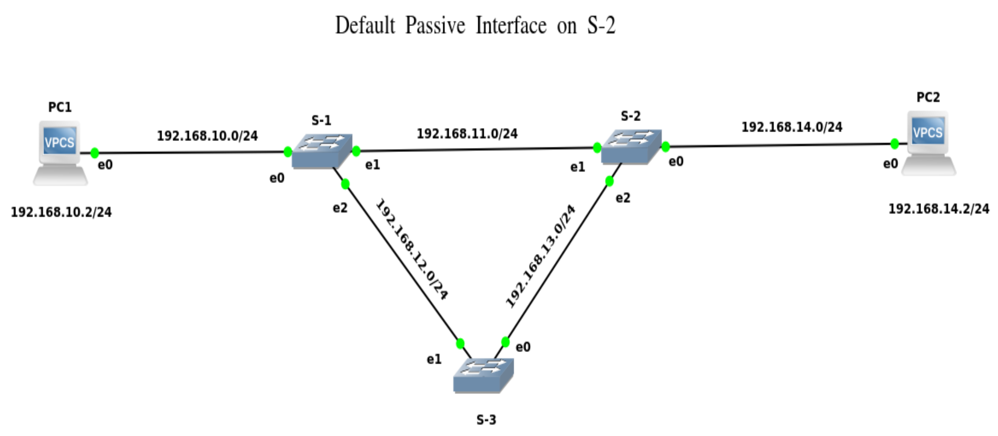
This section explains the step-by-step procedure to test OSPF Passive Interface and configure its features by running related commands in SONiC CLI.

Testbed Setup

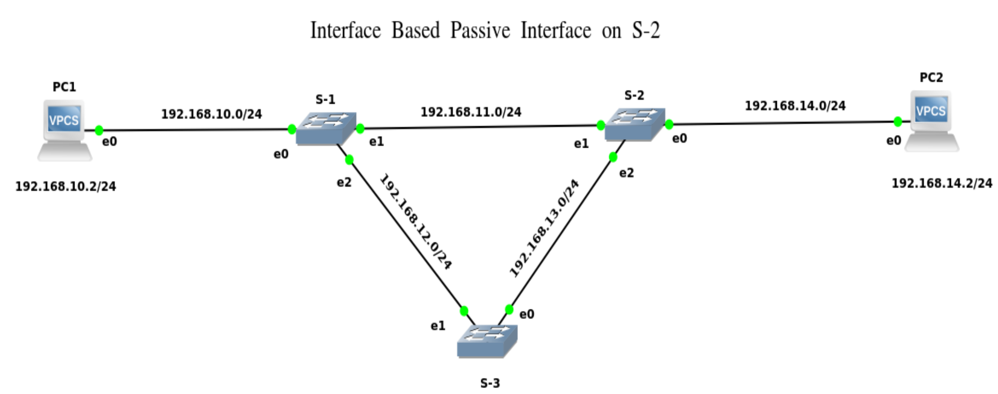
Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.



Fig#1: Network Topology



Fig#2: Network Topology

For each topology shown above, three switches (S-1, S-2 & S-3) and two hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using “sudo config reload” command.

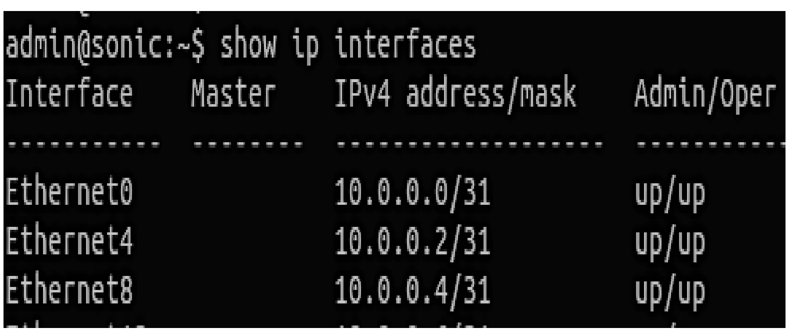
By following the above commands, make changes in "config_db” and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

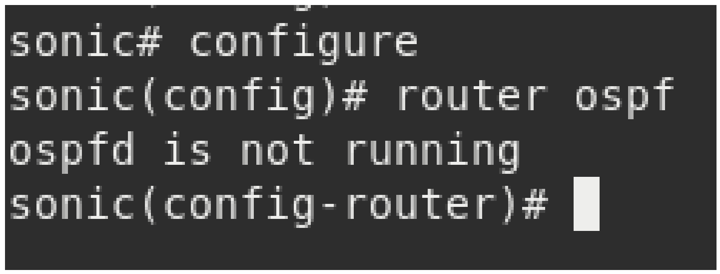
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

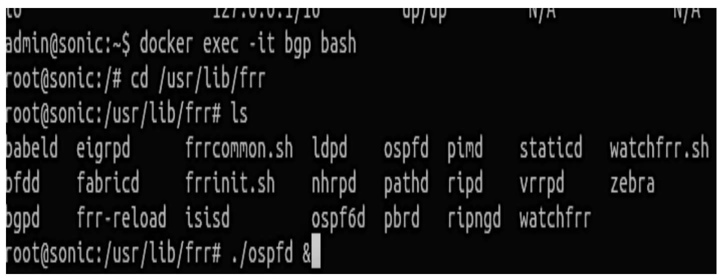
Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, the OSPF daemon is not running. For ospfd to be in the running mode use the following commands given below:

docker exec -it bgp bash
cd usr/lib/frr
ls
./ospfd &

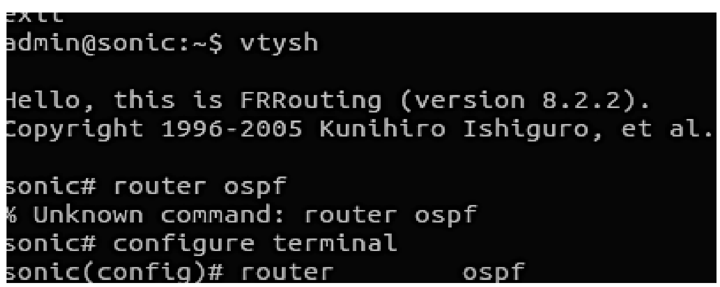




Step-5

OSPF must be enabled before carrying out any of the OSPF commands by using the commands given below:

vtysh
configure
router ospf



Step-6

In first topology (Fig#1), OSPF must be configured on a “network basis”. To configure OSPF, use the following command given below:

network <network address> area <address>

```
sonic(config-router)# router  ospf
sonic(config-router)# network 192.168.10.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.11.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.12.0/24 area 0.0.0.0
sonic(config-router)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

To check neighborship status, use the following command given below:

Show ip ospf neighbor

```
sonic# show ip ospf neighbor

Neighbor ID     Pri State           Up Time
192.168.11.3      1 Full/Backup     8.435s
192.168.13.3      1 Full/Backup     3m41s
```

Note: Notice that, mixing both network commands (network) and interface commands (ip ospf) on the same router is not supported. If (ip ospf) is present, (network) commands will fail. It is highly recommended to save the configurations using the “write” command.

Step-7

Passive interface in OSPF allows the connected network of an interface to be advertised throughout the OSPF domain but stops the sending of hello packets. If no hello packets are sent out of an interface, then an adjacency cannot be formed and if we configure an OSPF-enabled interface as passive where an adjacency already exists, the adjacency will drop. The command which is given below makes every interface of S-2 passive.

passive-interface default

```
sonic# configure
sonic(config)# router ospf
sonic(config-router)# passive-interface default
  <cr>
sonic(config-router)# passive-interface default
sonic(config-router)# exit
sonic(config)# exit
sonic# write
```

Step-8

To check the status of passive interfaces, use the following command given below:

show ip ospf interface

```
Ethernet0 is up
  ifindex 19, MTU 9100 bytes, BW 10 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.12.3/24, Broadcast 192.168.12.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 192.168.13.3, Network Type BROADCAST, Cost: 10000
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.13.3 Interface Address 192.168.12.3/24
  No backup designated router on this network
  Multicast group memberships: <None>
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    No Hellos (Passive interface)
  Neighbor Count is 0, Adjacent neighbor count is 0
Ethernet4 is up
  ifindex 20, MTU 9100 bytes, BW 10 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.13.3/24, Broadcast 192.168.13.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 192.168.13.3, Network Type BROADCAST, Cost: 10000
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.13.3 Interface Address 192.168.13.3/24
  No backup designated router on this network
  Multicast group memberships: <None>
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    No Hellos (Passive interface)
  Neighbor Count is 0, Adjacent neighbor count is 0
```

Step-9

After making every interface passive, there will be no entries in the neighborship table. To check the status of the table, use the following command given below:

show ip ospf neighbor

```
sonic# show ip ospf neighbor

Neighbor ID     Pri State
```

Step-10

To set the interfaces back to their default, just use the following command given below:

no passive-interface default

```
sonic# configure
sonic(config)# router ospf
sonic(config-router)# no passive-interface default
sonic(config-router)# exit
sonic(config)# exit
sonic# write
```

Step-11

In the 2nd topology (Fig#2), configurations are done on a per-interface basis. To configure OSPF, use the following command given below:

interface <interface name>

ip ospf area AREA

```
sonic# configure
sonic(config)# router ospf
sonic(config-router)# ospf router-id 192.168.102.1
sonic(config-router)# interface Ethernet0
sonic(config-if)# ip ospf area 0
```

Step-12

To make a specific interface passive, use the following command given below:

ip ospf passive <interface address>

```
sonic(config)# router ospf
sonic(config-router)# interface Ethernet0
sonic(config-if)# ip ospf passive 192.168.14.1
sonic(config-if)# exit
sonic(config)# exit
sonic# write
```

```
Ethernet0 is up
  ifindex 19, MTU 9100 bytes, BW 10 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.12.3/24, Broadcast 192.168.12.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 192.168.13.3, Network Type BROADCAST, Cost: 10000
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.13.3 Interface Address 192.168.12.3/24
  No backup designated router on this network
  Multicast group memberships: <None>
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    No Hellos (Passive interface)
  Neighbor Count is 0, Adjacent neighbor count is 0
```

Step-13

To set the interface back to its default, just use the following command given below:

no ip ospf passive <interface address>

```
sonic(config)# router ospf
sonic(config-router)# interface Ethernet8
sonic(config-if)# no ip ospf passive 192.168.13.2
sonic(config-if)# exit
sonic(config)# exit
sonic# write
```

```
Ethernet8 is up
  ifindex 21, MTU 9100 bytes, BW 10 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.13.2/24, Broadcast 192.168.13.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 192.168.14.1, Network Type BROADCAST, Cost: 10000
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.14.1 Interface Address 192.168.13.2/24
  Backup Designated Router (ID) 192.168.13.3, Interface Address 192.168.13.3
  Saved Network-LSA sequence number 0x80000005
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 9.213s
  Neighbor Count is 1, Adjacent neighbor count is 1
```

Step-14

Assign IP addresses to hosts given in network topology.

```
PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

When a passive interface is enabled on every interface of S-2 on a "network basis", the ping request sent by host PC1 to S-2 fails.

```
PC1> ping 192.168.11.3

192.168.11.3 icmp_seq=1 timeout
192.168.11.3 icmp_seq=2 timeout
192.168.11.3 icmp_seq=3 timeout
192.168.11.3 icmp_seq=4 timeout
192.168.11.3 icmp_seq=5 timeout
```

```
PC1> ping 192.168.13.2

192.168.13.2 icmp_seq=1 timeout
192.168.13.2 icmp_seq=2 timeout
192.168.13.2 icmp_seq=3 timeout
192.168.13.2 icmp_seq=4 timeout
192.168.13.2 icmp_seq=5 timeout
```

Since (192.168.13.3) is a S-3 interface and it is not passive, traffic is reaching here which can be seen below:

```
PC1> ping 192.168.13.3

84 bytes from 192.168.13.3 icmp_seq=1 ttl=63 time=6.585 ms
84 bytes from 192.168.13.3 icmp_seq=2 ttl=63 time=6.320 ms
84 bytes from 192.168.13.3 icmp_seq=3 ttl=63 time=6.172 ms
84 bytes from 192.168.13.3 icmp_seq=4 ttl=63 time=6.311 ms
84 bytes from 192.168.13.3 icmp_seq=5 ttl=63 time=6.760 ms
```

On the other hand, when the switch is configured on a "per interface" basis, ping requests are sent to every interface will fail only if all the interfaces are passive. If one of the interfaces is not passive, then the ping request will be successful and sent to every interface.

OSPF Totally Stub Area

Introduction

This section explains the step-by-step procedure to test OSPF Totally Stub Area and configure its features by running related commands in SONiC CLI.

Testbed Setup

Please refer to [Testbed]() section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

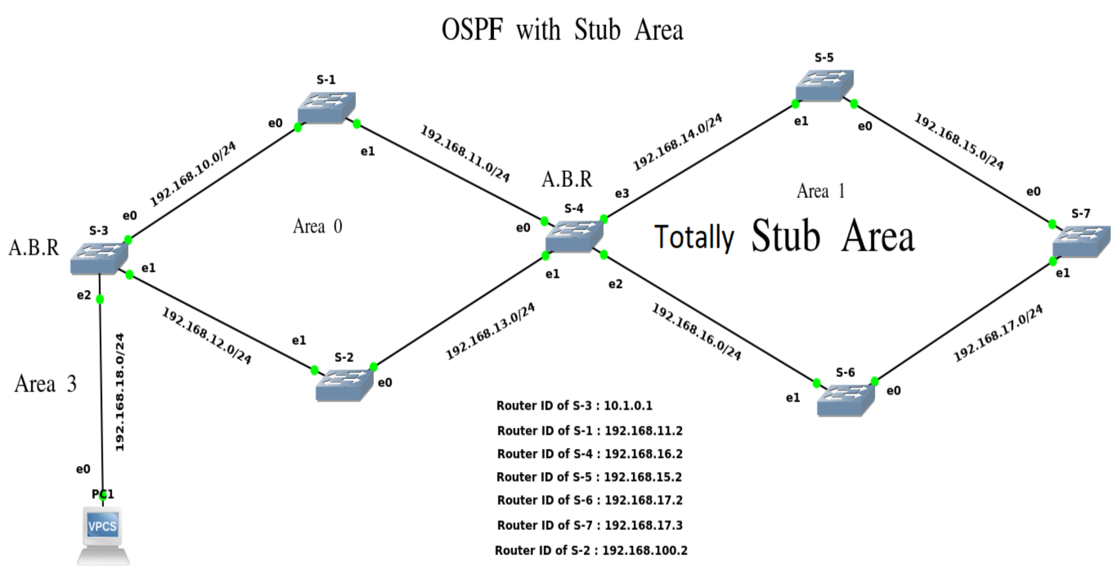


Fig: Network Topology

For the above topology, seven switches (S-1 to S-7) and one host is used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A     fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A     fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A     fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A    fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A    fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A    fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A    fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A    fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A    fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A    fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A    fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A    fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A    fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A    fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A    fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A    fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A    fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using "sudo config reload" command.

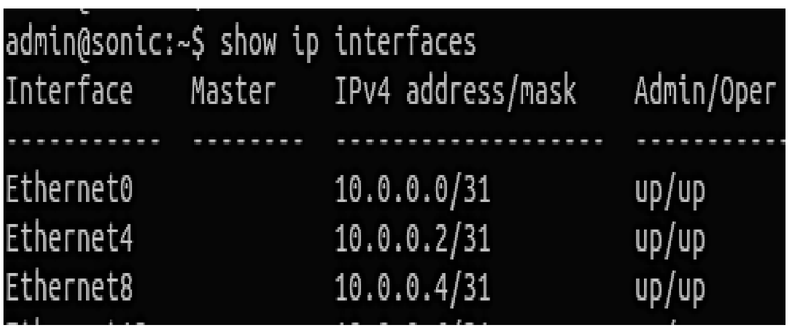
By following the above commands, make changes in "config_db" and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

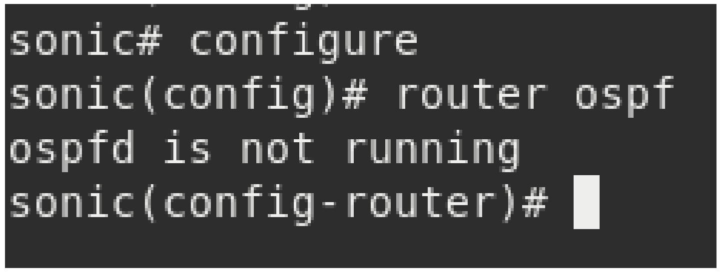
- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

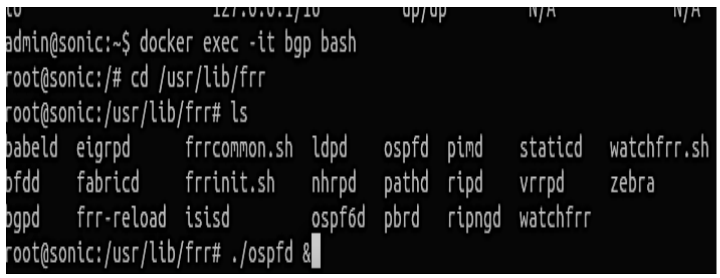
Step-4

By default, the OSPF daemon is not running. For ospfd to be in the running mode use the following commands given below:

docker exec -it bgp bash
cd usr/lib/frr
ls
./ospfd &



```
sonic# configure
sonic(config)# router ospf
ospfd is not running
sonic(config-router)#
```

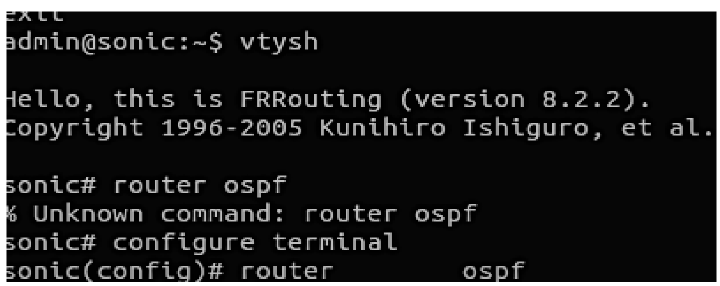


```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd /usr/lib/frr
root@sonic:/usr/lib/frr# ls
babeld  eigrpd     frrcommon.sh  ldpd    ospfd  pimd    staticd   watchfrr.sh
bfdd    fabricd    frrinit.sh    nhrpd   pathd  ripd    vrrpd     zebra
bgpd    frr-reload isisd         ospf6d  pbrd   ripngd  watchfrr
root@sonic:/usr/lib/frr# ./ospfd &
```

Step-5

OSPF must be enabled before carrying out any of the OSPF commands by using the commands given below:

vtysh
configure
router ospf



```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# router ospf
% Unknown command: router ospf
sonic# configure terminal
sonic(config)# router        ospf
```

Step-6

In first topology (Fig#1), OSPF must be configured on a “network basis”. To configure OSPF, use the following command given below:

network <network address> area <address>

```
sonic(config-router)# router  ospf
sonic(config-router)# network 192.168.10.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.11.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.12.0/24 area 0.0.0.0
sonic(config-router)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: Notice that, mixing both network commands (network) and interface commands (ip ospf) on the same router is not supported. If (ip ospf) is present, (network) commands will fail. It is highly recommended to save the configurations using the “write” command.

Step-7

A totally stubby area (TSA) is a stub area in which summary link-state advertisement (type 3 LSAs) are not sent. A default summary LSA, with a prefix of 0.0.0.0/0 is originated into the stub area by an ABR, so that devices in the area can forward all traffic for which a specific route is not known, via ABR. In our topology “Area 1” is Totally Stub Area. Apply the following command on S-4 to S-7 given below:

area A.B.C.D stub

```
sonic# configure
sonic(config)# router ospf
sonic(config-router)# area 1 stub
sonic(config-router)# exit
sonic(config)# exit
sonic# write
```

Step-8

There is another command which must be configured only on Area Border Router (ABR) which is given below:

area A.B.C.D stub no-summary

```
sonic(config)# router ospf
sonic(config-router)# area 1 stub no-summary
sonic(config-router)# exit
sonic(config)# exit
sonic# write
```

Step-9

To check the status of totally stub area, use the following command given below:

show ip ospf

```
Number of areas attached to this router: 1
Area ID: 0.0.0.1 (Stub)
  Shortcutting mode: Default, S-bit consensus: no
  Number of interfaces in this area: Total: 2, Active: 2
  Number of fully adjacent neighbors in this area: 2
  Area has no authentication
  Number of full virtual adjacencies going through this area: 0
  SPF algorithm executed 27 times
```

Step-10

To check the status of routes, just use the following command given below:

show ip route

```
O>* 192.168.10.0/24 [110/30000] via 192.168.14.2, Ethernet4, weight 1,
O>* 192.168.11.0/24 [110/20000] via 192.168.14.2, Ethernet4, weight 1,
O>* 192.168.12.0/24 [110/30000] via 192.168.14.2, Ethernet4, weight 1,
O>* 192.168.13.0/24 [110/20000] via 192.168.14.2, Ethernet4, weight 1,
O   192.168.14.0/24 [110/10000] is directly connected, Ethernet4, weig
C>* 192.168.14.0/24 is directly connected, Ethernet4, 00:14:43
O   192.168.15.0/24 [110/10000] is directly connected, Ethernet0, weig
C>* 192.168.15.0/24 is directly connected, Ethernet0, 00:14:43
O>* 192.168.16.0/24 [110/20000] via 192.168.14.2, Ethernet4, weight 1,
O>* 192.168.17.0/24 [110/20000] via 192.168.15.3, Ethernet0, weight 1,
O>* 192.168.18.0/24 [110/40000] via 192.168.14.2, Ethernet4, weight 1,
sonic#
```

```
O   192.168.14.0/24 [110/10000] is directly connected, Ethernet4, weight 1, 00:17:35
C>* 192.168.14.0/24 is directly connected, Ethernet4, 00:29:52
O   192.168.15.0/24 [110/10000] is directly connected, Ethernet0, weight 1, 00:08:38
C>* 192.168.15.0/24 is directly connected, Ethernet0, 00:29:52
O>* 192.168.16.0/24 [110/20000] via 192.168.14.2, Ethernet4, weight 1, 00:09:45
O>* 192.168.17.0/24 [110/20000] via 192.168.15.3, Ethernet0, weight 1, 00:08:36
```

The above figure shows a comparison between routes of S-5 before and after configuration. Before making “Area 1” a totally stub area, S-5 shows inter routes as well. But after making “Area 1” a totally stub area, S-5 does not show routes of the backbone area.

Step-11

Assign IP addresses to hosts given in network topology.

```
PC1> ip 192.168.10.2/24 192.168.10.1
Checking for duplicate address...
PC1 : 192.168.10.2 255.255.255.0 gateway 192.168.10.1

PC1> save
Saving startup configuration to startup.vpc
.  done

PC1> write
Saving startup configuration to startup.vpc
.  done
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

By making “Area 1” a totally stub area, it does not send type 3 LSAs. A default summary LSA, with a prefix of 0.0.0.0/0 is originated into the stub area by an ABR, so that devices in the area can forward all traffic for which a specific route is not known, via ABR. Routers of totally stub area do not show internal routes.

```
sonic# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backu
       t - trapped, o - offload failure

O>* 0.0.0.0/0 [110/10001] via 192.168.16.2, Ethernet4, weight 1, 00:12:56
```

OSPF Virtual Link

Introduction

This section explains the step-by-step procedure to test OSPF Virtual Link and configure its features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches and hosts.

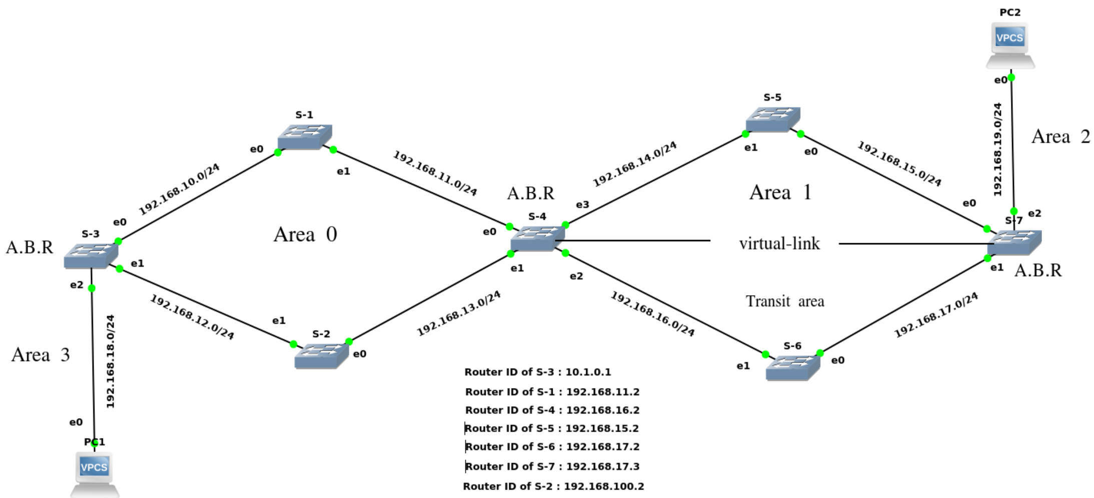


Fig: Network Topology

For the above topology, seven switches (S-1 to S-7) and two hosts are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

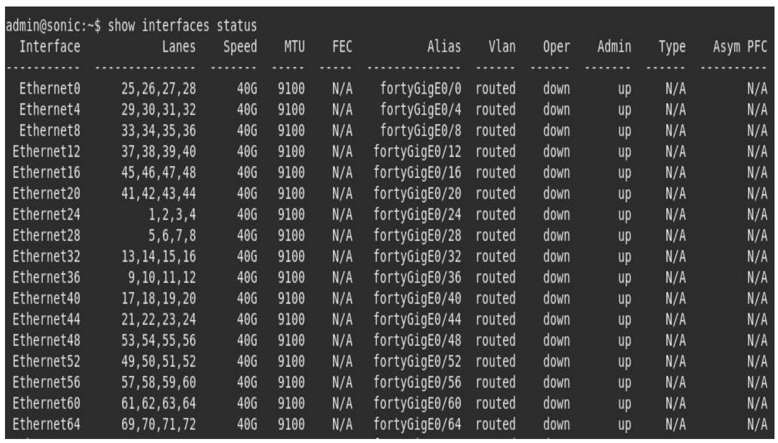
For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status



admin@sonic:~$ show interfaces status

Interface	Lanes	Speed	MTU	FEC	Alias	Vlan	Oper	Admin	Type	Asym PFC
Ethernet0	25,26,27,28	40G	9100	N/A	fortyGigE0/0	routed	down	up	N/A	N/A
Ethernet4	29,30,31,32	40G	9100	N/A	fortyGigE0/4	routed	down	up	N/A	N/A
Ethernet8	33,34,35,36	40G	9100	N/A	fortyGigE0/8	routed	down	up	N/A	N/A
Ethernet12	37,38,39,40	40G	9100	N/A	fortyGigE0/12	routed	down	up	N/A	N/A
Ethernet16	45,46,47,48	40G	9100	N/A	fortyGigE0/16	routed	down	up	N/A	N/A
Ethernet20	41,42,43,44	40G	9100	N/A	fortyGigE0/20	routed	down	up	N/A	N/A
Ethernet24	1,2,3,4	40G	9100	N/A	fortyGigE0/24	routed	down	up	N/A	N/A
Ethernet28	5,6,7,8	40G	9100	N/A	fortyGigE0/28	routed	down	up	N/A	N/A
Ethernet32	13,14,15,16	40G	9100	N/A	fortyGigE0/32	routed	down	up	N/A	N/A
Ethernet36	9,10,11,12	40G	9100	N/A	fortyGigE0/36	routed	down	up	N/A	N/A
Ethernet40	17,18,19,20	40G	9100	N/A	fortyGigE0/40	routed	down	up	N/A	N/A
Ethernet44	21,22,23,24	40G	9100	N/A	fortyGigE0/44	routed	down	up	N/A	N/A
Ethernet48	53,54,55,56	40G	9100	N/A	fortyGigE0/48	routed	down	up	N/A	N/A
Ethernet52	49,50,51,52	40G	9100	N/A	fortyGigE0/52	routed	down	up	N/A	N/A
Ethernet56	57,58,59,60	40G	9100	N/A	fortyGigE0/56	routed	down	up	N/A	N/A
Ethernet60	61,62,63,64	40G	9100	N/A	fortyGigE0/60	routed	down	up	N/A	N/A
Ethernet64	69,70,71,72	40G	9100	N/A	fortyGigE0/64	routed	down	up	N/A	N/A

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using “sudo config reload” command.

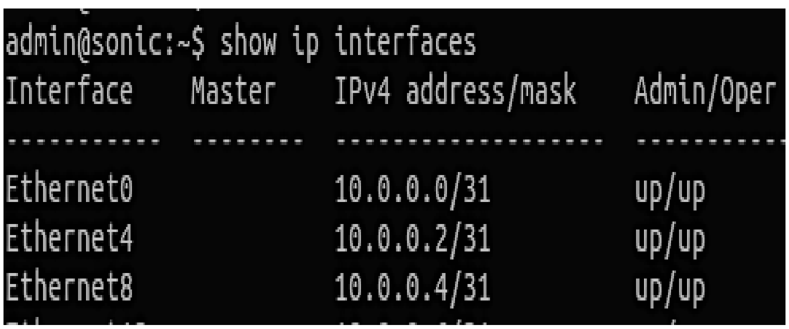
By following the above commands, make changes in "config_db” and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

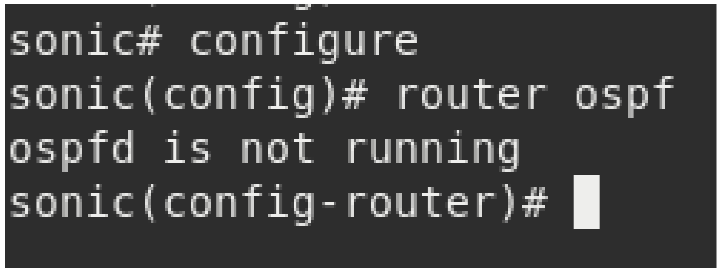
By default, the OSPF daemon is not running. For ospfd to be in the running mode use the following commands given below:

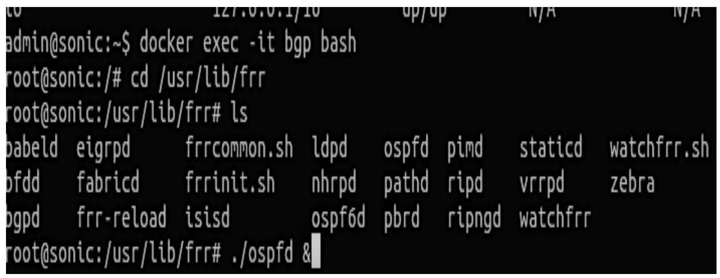
docker exec -it bgp bash

cd usr/lib/frr

ls

./ospfd &





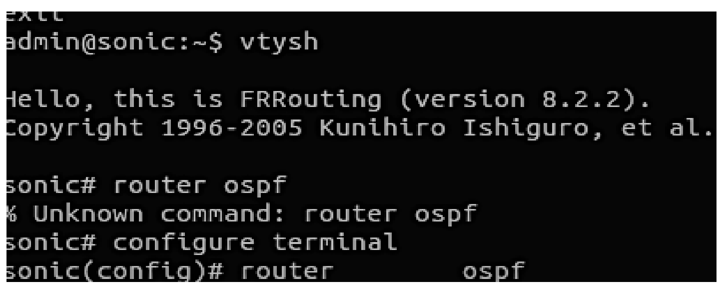
Step-5

OSPF must be enabled before carrying out any of the OSPF commands by using the commands given below:

vtysh

configure

router ospf



Step-6

After enabling OSPF, assign all the network addresses to which the switch is directly connected by using the command given below:

network <network address> area <address>

```
sonic(config)# router      ospf
sonic(config-router)# router  ospf
sonic(config-router)# network 192.168.10.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.11.0/24 area 0.0.0.0
sonic(config-router)# network 192.168.12.0/24 area 0.0.0.0
sonic(config-router)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-7

The backbone area forms the central hub of an OSPF network. All other areas are connected to it, and inter-area routing happens via routers connected to the backbone area and to their own non-backbone areas. The backbone area distributes all routing information between the non-backbone areas. The backbone area is always assigned area ID 0.0.0.0 (All area IDs must be unique within an AS.) All other networks or areas in the AS must be directly connected to the backbone area by a router that has interfaces in more than one area.

- To check the Area ID of an OSPF, use the following command given below:
  show ip ospf

```
Area ID: 0.0.0.0 (Backbone)
  Number of interfaces in this area: Total: 2, Active: 2
  Number of fully adjacent neighbors in this area: 2
  Area has no authentication
  SPF algorithm executed 2370 times
  Number of LSA 19
  Number of router LSA 5. Checksum Sum 0x00019bf9
  Number of network LSA 4. Checksum Sum 0x00025f76
  Number of summary LSA 10. Checksum Sum 0x00047540
  Number of ASBR summary LSA 0. Checksum Sum 0x00000000
  Number of NSSA LSA 0. Checksum Sum 0x00000000
  Number of opaque link LSA 0. Checksum Sum 0x00000000
  Number of opaque area LSA 0. Checksum Sum 0x00000000
```

A router with interfaces in two (or more) different areas is an **area border router**. An area border router is in the OSPF boundary between two areas. In our topology S-3, S-4 & S-7 are ABRs.

- To check ABR in an OSPF, use the following command given below:
  show ip ospf

```
sonic# show ip ospf
 OSPF Routing Process, Router ID: 10.1.0.1
 Supports only single TOS (TOS0) routes
 This implementation conforms to RFC2328
 RFC1583Compatibility flag is disabled
 OpaqueCapability flag is disabled
 Initial SPF scheduling delay 0 millisec(s)
 Minimum hold time between consecutive SPFs 50 millisec(s)
 Maximum hold time between consecutive SPFs 5000 millisec(s)
 Hold time multiplier is currently 1
 SPF algorithm last executed 14m42s ago
 Last SPF duration 0.005s
 SPF timer is inactive
 LSA minimum interval 5000 msecs
 LSA minimum arrival 1000 msecs
 Write Multiplier set to 20
 Refresh timer 10 secs
 Maximum multiple paths(ECMP) supported  256
 This router is an ABR, ABR type is: Alternative Cisco
 Number of external LSA 0. Checksum Sum 0x00000000
 Number of opaque AS LSA 0. Checksum Sum 0x00000000
```

Transit area is the area that has a virtual link connecting two or more ABRs attached to this area. Thus, having a virtual link provisioned across the area is the necessary thing to make the area transit. In our topology, the transit area is between two ABRs S-4 and S-7.

Step-8

Virtual Links are used for connecting areas to the Backbone Area, that are not directly connected to the Backbone Area. For this job, there is one Transit Area that helps to connect the area to the Backbone Area. Both ends of the Transit Area is configured with Virtual Links.

- To configure the virtual link in ABR, use the following command given below:
  area A.B.C.D virtual-link A.B.C.D

In Area address, give that ID which acts as a transit area. In our topology, Area 1 acts as a transit area. In Virtual Link ID, an address is to be provided which must be Router ID. In our topology, S-4 and S-7 are part of the transit area. There is an important thing while configuring ABRs for creating a virtual link in the transit area. While configuring S-4, advertise the Router ID of S-7 in S-4. In the same way, advertise the Router ID of S-4 in S-7 while configuring S-7.

```
sonic(config-router)# area 1 virtual-link 192.168.17.3
sonic(config-router)# exit
sonic(config)# exit
sonic#
```

- To check the status of the virtual link, use the following command given below:
  show ip ospf neighbour

```
onic# show ip ospf neighbor

eighbor ID     Pri State           Up Time         Dead Time Address         Interface
92.168.11.2      1 Full/DR         3h07m16s          34.422s 192.168.11.2    Ethernet0:192.168.11.3
92.168.100.2     1 Full/Backup     28m32s            37.938s 192.168.13.2    Ethernet4:192.168.13.3
92.168.15.2      1 Full/Backup     3h03m02s          33.097s 192.168.14.3    Ethernet12:192.168.14.2
92.168.17.2      1 Full/Backup     3h01m14s          35.594s 192.168.16.3    Ethernet8:192.168.16.2
92.168.17.3      1 Full/DROther    2h39m20s          30.741s 192.168.15.3    VLINK2
```

In the above figure, the command is executed in S-4. In the neighbor ID column, the Router ID of S-7 is displayed which is 192.168.17.3. In the above figure, it is clearly shown that the virtual link is created between S-4 and S-7 (VLINK2).

```
sonic# show ip ospf neighbor

Neighbor ID     Pri State           Up Time         Dead Time Address         Interface
192.168.15.2      1 Full/DR         3h02m28s          32.435s 192.168.15.2    Ethernet0:192.168.15.3
192.168.17.2      1 Full/DR         3h02m23s          32.554s 192.168.17.2    Ethernet4:192.168.17.3
192.168.16.2      1 Full/DROther    2h41m57s          38.342s 192.168.14.2    VLINK2
```

The above figure displays result of S-7.

Step-9

Repeat steps 1-8 for other switches as well.

Step-10

Check OSPF status using the commands given below:

show ip ospf

show ip ospf database router <router id

```
sonic# show ip ospf
 OSPF Routing Process, Router ID: 192.168.12.2
 Supports only single TOS (TOS0) routes
 This implementation conforms to RFC2328
 RFC1583Compatibility flag is disabled
 OpaqueCapability flag is disabled
 Initial SPF scheduling delay 0 millisec(s)
 Minimum hold time between consecutive SPFs 50 millisec(s)
 Maximum hold time between consecutive SPFs 5000 millisec(s)
 Hold time multiplier is currently 1
 SPF algorithm last executed 27m52s ago
 Last SPF duration 518 usecs
 SPF timer is inactive
 LSA minimum interval 5000 msecs
 LSA minimum arrival 1000 msecs
 Write Multiplier set to 20
 Refresh timer 10 secs
 Maximum multiple paths(ECMP) supported  256
 Number of external LSA 0. Checksum Sum 0x00000000
 Number of opaque AS LSA 0. Checksum Sum 0x00000000
 Number of areas attached to this router: 1
 Area ID: 0.0.0.0 (Backbone)
```

```
sonic# show ip ospf database router 192.168.12.2

       OSPF Router with ID (192.168.12.2)


                Router Link States (Area 0.0.0.0)

  LS age: 54
  Options: 0x2  : *|-|-|-|-|-|E|-
  LS Flags: 0x3
  Flags: 0x0
  LS Type: router-LSA
  Link State ID: 192.168.12.2
  Advertising Router: 192.168.12.2
  LS Seq Number: 8000000c
  Checksum: 0x87be
  Length: 60

   Number of Links: 3

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.11.2
     (Link Data) Router Interface address: 192.168.11.2
      Number of TOS metrics: 0
       TOS 0 Metric: 10000

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.12.2
```

Step-11

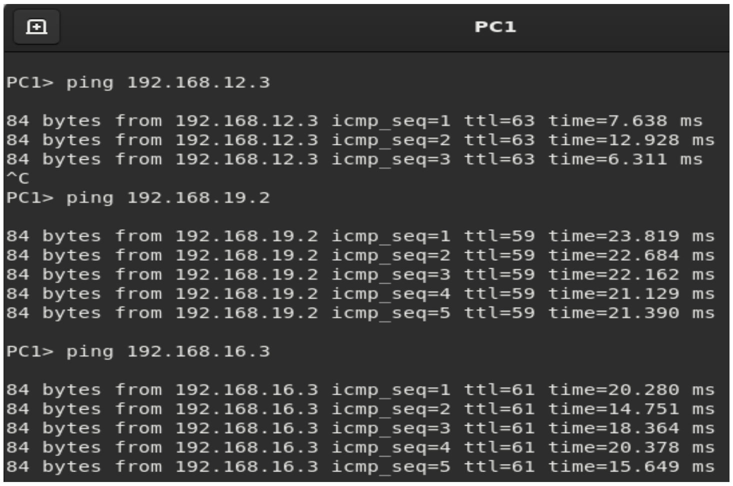
Assign IP addresses to hosts given in network topology.

```
PC2> ip 192.168.19.2 192.168.19.1
Checking for duplicate address...
PC2 : 192.168.19.2 255.255.255.0 gateway 192.168.19.1

PC2> save
Saving startup configuration to startup.vpc
.  done

PC2>
```

Note: It is highly recommended to save configurations in the host using the “save” command.

Result

After configuring the switches and hosts, hosts can send traffic. In the figure below, it is clearly seen that host PC1 can send traffic to PC2 and other networks. So, a virtual link is successfully configured in the topology.



```
PC1

PC1> ping 192.168.12.3

84 bytes from 192.168.12.3 icmp_seq=1 ttl=63 time=7.638 ms
84 bytes from 192.168.12.3 icmp_seq=2 ttl=63 time=12.928 ms
84 bytes from 192.168.12.3 icmp_seq=3 ttl=63 time=6.311 ms
^C
PC1> ping 192.168.19.2

84 bytes from 192.168.19.2 icmp_seq=1 ttl=59 time=23.819 ms
84 bytes from 192.168.19.2 icmp_seq=2 ttl=59 time=22.684 ms
84 bytes from 192.168.19.2 icmp_seq=3 ttl=59 time=22.162 ms
84 bytes from 192.168.19.2 icmp_seq=4 ttl=59 time=21.129 ms
84 bytes from 192.168.19.2 icmp_seq=5 ttl=59 time=21.390 ms

PC1> ping 192.168.16.3

84 bytes from 192.168.16.3 icmp_seq=1 ttl=61 time=20.280 ms
84 bytes from 192.168.16.3 icmp_seq=2 ttl=61 time=14.751 ms
84 bytes from 192.168.16.3 icmp_seq=3 ttl=61 time=18.364 ms
84 bytes from 192.168.16.3 icmp_seq=4 ttl=61 time=20.378 ms
84 bytes from 192.168.16.3 icmp_seq=5 ttl=61 time=15.649 ms
```

Testing BGP Views in SONiC

Introduction

This section explains the step-by-step procedure to test BGP Views and configure its features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches.

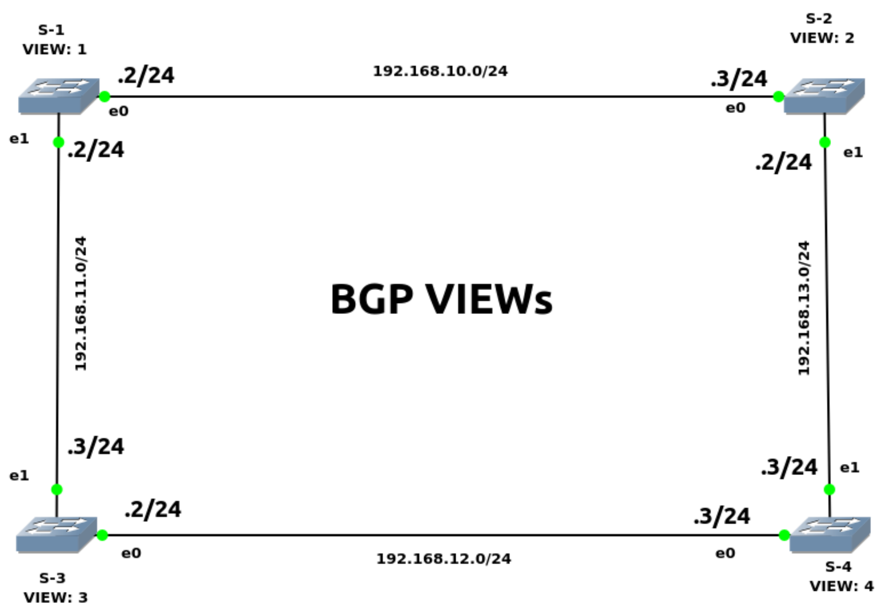


Fig: Network Topology

For the topology shown above, four switches (S-1, S-2, S-3& S-4) are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC          Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  -------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A   fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A   fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A   fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A  fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A  fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A  fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A  fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A  fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A  fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A  fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A  fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A  fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A  fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A  fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A  fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A  fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A  fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using "sudo config reload" command.

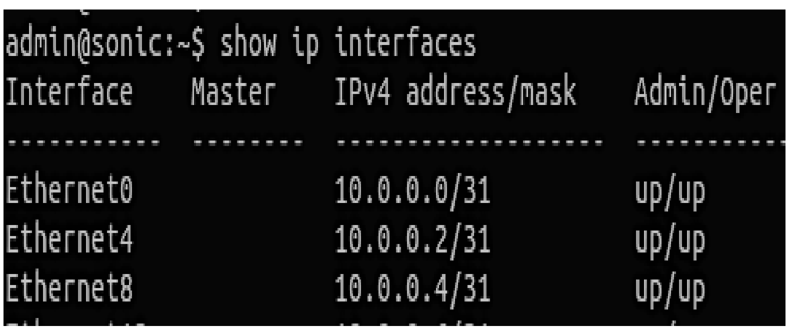
By following the above commands, make changes in "config_db" and then save it.

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces



Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

BGP views are almost the same as normal BGP processes, except that routes selected by BGP are not installed into the kernel routing table. By default, the BGP is in running mode with Autonomous System Number (ASN) 65100. To configure a normal BGP process, first BGP process (65100) must be stopped, and then configurations are made while in the case of BGP Views there is no need of stopping the BGP process (65100). To configure BGP View on S-1, use the following commands given below:

vtysh

configure

router bgp AS-NUMBER view NAME

One can use an arbitrary word for the NAME.

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router bgp 1 view 1
sonic(config-router)# exit
sonic(config)# router bgp 1000
BGP is already running; AS is 65100
sonic(config)# router bgp 1 view 1
```

Step-5

Now form a connection with neighbors by using the following command given below:

neighbor <Interface address> remote-as <ASN>

```
sonic(config)# router bgp 1 view 1
sonic(config-router)# neighbor 192.168.10.3 remote-as 2
sonic(config-router)# neighbor 192.168.11.3 remote-as 3
sonic(config-router)# exit
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-6

Repeat steps 1-5 for the other switches.

Step-7

To check the BGP summary, use the following command given below:

show ip bgp summary

```
sonic# show ip bgp summary

IPv4 Unicast Summary (VRF default):
BGP router identifier 10.1.0.1, local AS number 65100 vrf-id 0
BGP table version 1
RIB entries 1, using 184 bytes of memory
Peers 29, using 20 MiB of memory
Peer groups 2, using 128 bytes of memory

Neighbor        V         AS   MsgRcvd   MsgSent   TblVer  InQ OutQ  Up/Down State/PfxRcd   PfxSnt Desc
10.0.0.7        4      65200         0         0        0    0    0    never       Active        0 ARISTA04T2
10.0.0.9        4      65200         0         0        0    0    0    never       Active        0 ARISTA05T2
10.0.0.11       4      65200         0         0        0    0    0    never       Active        0 ARISTA06T2
10.0.0.13       4      65200         0         0        0    0    0    never       Active        0 ARISTA07T2
10.0.0.15       4      65200         0         0        0    0    0    never       Active        0 ARISTA08T2
10.0.0.17       4      65200         0         0        0    0    0    never       Active        0 ARISTA09T2
10.0.0.19       4      65200         0         0        0    0    0    never       Active        0 ARISTA10T2
10.0.0.21       4      65200         0         0        0    0    0    never       Active        0 ARISTA11T2
10.0.0.23       4      65200         0         0        0    0    0    never       Active        0 ARISTA12T2
10.0.0.25       4      65200         0         0        0    0    0    never       Active        0 ARISTA13T2
10.0.0.27       4      65200         0         0        0    0    0    never       Active        0 ARISTA14T2
10.0.0.29       4      65200         0         0        0    0    0    never       Active        0 ARISTA15T2
10.0.0.31       4      65200         0         0        0    0    0    never       Active        0 ARISTA16T2
10.0.0.33       4      64001         0         0        0    0    0    never       Active        0 ARISTA01T0
10.0.0.35       4      64002         0         0        0    0    0    never       Active        0 ARISTA02T0
10.0.0.37       4      64003         0         0        0    0    0    never       Active        0 ARISTA03T0
10.0.0.39       4      64004         0         0        0    0    0    never       Active        0 ARISTA04T0
10.0.0.41       4      64005         0         0        0    0    0    never       Active        0 ARISTA05T0
10.0.0.43       4      64006         0         0        0    0    0    never       Active        0 ARISTA06T0
10.0.0.45       4      64007         0         0        0    0    0    never       Active        0 ARISTA07T0
10.0.0.47       4      64008         0         0        0    0    0    never       Active        0 ARISTA08T0
10.0.0.49       4      64009         0         0        0    0    0    never       Active        0 ARISTA09T0
10.0.0.51       4      64010         0         0        0    0    0    never       Active        0 ARISTA10T0
10.0.0.53       4      64011         0         0        0    0    0    never       Active        0 ARISTA11T0
10.0.0.55       4      64012         0         0        0    0    0    never       Active        0 ARISTA12T0
10.0.0.57       4      64013         0         0        0    0    0    never       Active        0 ARISTA13T0
10.0.0.59       4      64014         0         0        0    0    0    never       Active        0 ARISTA14T0
10.0.0.61       4      64015         0         0        0    0    0    never       Active        0 ARISTA15T0
10.0.0.63       4      64016         0         0        0    0    0    never       Active        0 ARISTA16T0

Total number of neighbors 29
```

From the above figure, it is confirmed that routes selected by the view are not installed into the kernel routing table. In the above figure, all the neighbors are default configured.

Step-8

To check the status of BGP neighbors, use the following command given below:

show bgp neighbors

```
sonic# show bgp neighbors
BGP neighbor is 10.0.0.7, remote AS 65200, local AS 65100, external link
 Description: ARISTA04T2
 Member of peer-group PEER_V4 for session parameters
  BGP version 4, remote router ID 0.0.0.0, local router ID 10.1.0.1
  BGP state = Active
  Last read 00:51:53, Last write never
  Hold time is 180, keepalive interval is 60 seconds
  Graceful restart information:
    Local GR Mode: Helper*
    Remote GR Mode: NotApplicable
    R bit: False
    Timers:
      Configured Restart Time(sec): 120
      Received Restart Time(sec): 0
  Message statistics:
    Inq depth is 0
    Outq depth is 0
                         Sent       Rcvd
    Opens:                  0          0
    Notifications:          0          0
    Updates:                0          0
    Keepalives:             0          0
    Route Refresh:          0          0
    Capability:             0          0
    Total:                  0          0
  Minimum time between advertisement runs is 0 seconds
```

From the above figure, it can be shown that there is no entry of neighbors configured in the topology.

Step-9

To view the BGP configuration file of S-1, use the following commands given below:

docker exec -it bgp bash
cd /etc/frr
ls
cat bgpd.conf

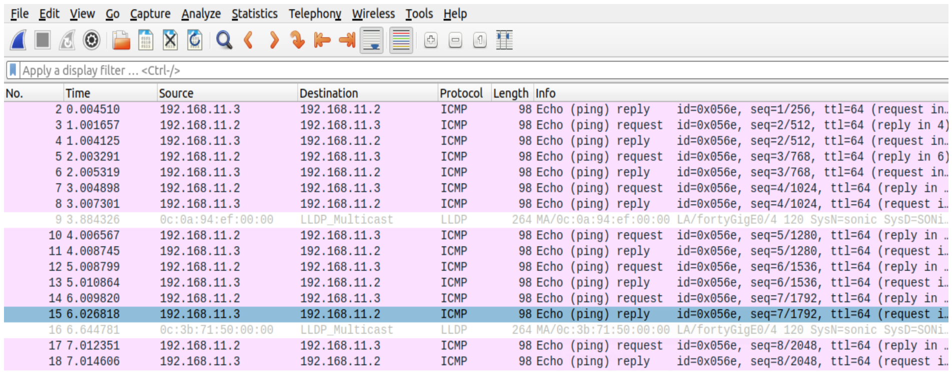
```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd /etc/frr/bgpd.conf
bash: cd: /etc/frr/bgpd.conf: Not a directory
root@sonic:/# cd /etc/frr
root@sonic:/etc/frr# ls
bgpd.conf      staticd.conf      vtysh.conf  zebra.conf.sav
bgpd.conf.sav  staticd.conf.sav  zebra.conf
root@sonic:/etc/frr# cat bgpd.conf
```

```
exit
!
router bgp 2 view 2
 neighbor 192.168.10.2 remote-as 1
 neighbor 192.168.13.3 remote-as 4
!
exit
!
ip prefix-list PL_LoopbackV4 seq 5 permit 10.1.0.1/32
!
```

From the above figure it can be shown that neighbor entries are only present in the configuration file.

Step-10

For analysis of BGP View, Wireshark has been used whose result is shown below:

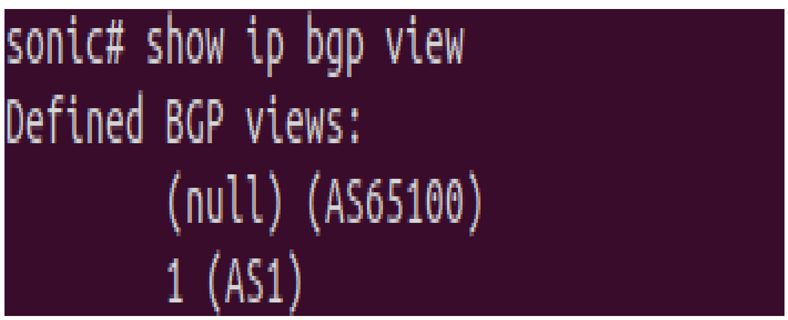


In the above figure, there is no entry of BGP protocol.

Step-11

To check the BGP view status, use the following command given below:

show ip bgp view



In FRR Document, there is a command given to check the status of BGP View but it does not show any result. That command is given below:

show [ip] bgp view NAME

Result

When a ping request is generated from S-1 to S-2 and S-1 to S-3, a successful reply shows that BGP View is successfully configured in the topology.

```
sonic# ping 192.168.10.3
PING 192.168.10.3 (192.168.10.3) 56(84) bytes of data.
64 bytes from 192.168.10.3: icmp_seq=1 ttl=64 time=6.38 ms
64 bytes from 192.168.10.3: icmp_seq=2 ttl=64 time=4.40 ms
64 bytes from 192.168.10.3: icmp_seq=3 ttl=64 time=6.67 ms
64 bytes from 192.168.10.3: icmp_seq=4 ttl=64 time=4.83 ms
64 bytes from 192.168.10.3: icmp_seq=5 ttl=64 time=4.13 ms
64 bytes from 192.168.10.3: icmp_seq=6 ttl=64 time=4.49 ms
64 bytes from 192.168.10.3: icmp_seq=7 ttl=64 time=4.29 ms
^C
--- 192.168.10.3 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 13ms
rtt min/avg/max/mdev = 4.131/5.026/6.670/0.970 ms
sonic# ping 192.168.11.3
PING 192.168.11.3 (192.168.11.3) 56(84) bytes of data.
64 bytes from 192.168.11.3: icmp_seq=1 ttl=64 time=5.13 ms
64 bytes from 192.168.11.3: icmp_seq=2 ttl=64 time=3.91 ms
64 bytes from 192.168.11.3: icmp_seq=3 ttl=64 time=4.10 ms
64 bytes from 192.168.11.3: icmp_seq=4 ttl=64 time=4.30 ms
64 bytes from 192.168.11.3: icmp_seq=5 ttl=64 time=4.21 ms
```

BGP Network Advertisement

Introduction

This section explains the step-by-step procedure to test BGP Network Advertisement and configure its features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches.

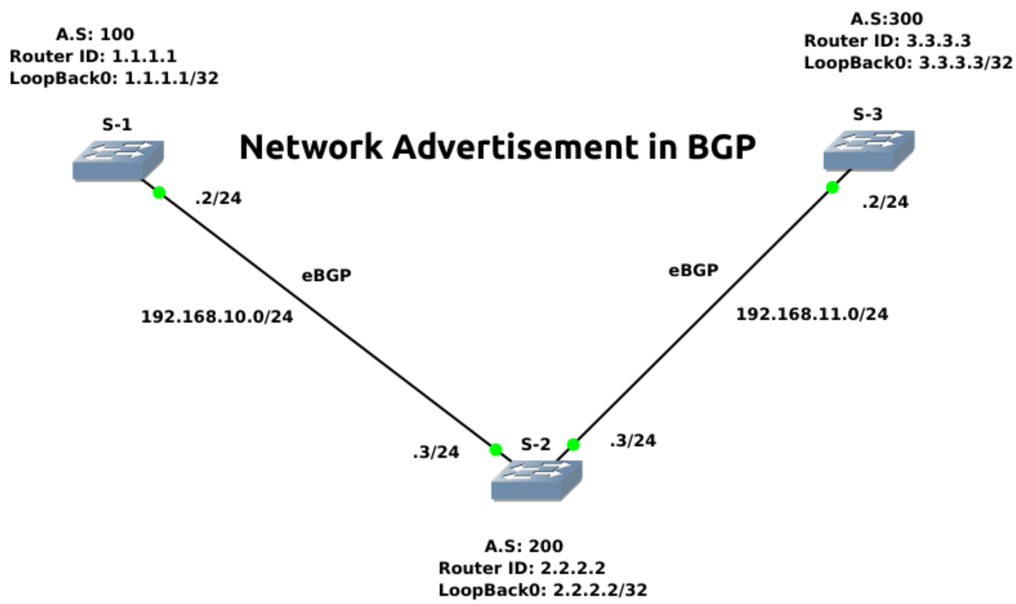


Fig: Network Topology

For the topology shown above, three switches (S-1, S-2 & S-3) are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC          Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  -------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A   fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A   fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A   fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A  fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A  fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A  fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A  fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A  fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A  fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36       9,10,11,12      40G   9100    N/A  fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A  fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A  fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A  fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A  fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A  fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A  fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A  fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring "config_db" and the path is /etc/sonic. To configure "config_db", command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using "sudo config reload" command.

By following the above commands, make changes in "config_db" and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
-- INSERT --
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces

```
admin@sonic:~$ show ip interfaces
Interface    Master    IPv4 address/mask    Admin/Oper
-----------  --------  -------------------  ----------
Ethernet0              10.0.0.0/31          up/up
Ethernet4              10.0.0.2/31          up/up
Ethernet8              10.0.0.4/31          up/up
```

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, the BGP is in running mode with Autonomous System Number (ASN) 65100. First, stop this BGP process using the commands given below:

vtysh

configure

no router bgp 65100

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router bgp 100
BGP is already running; AS is 65100
sonic(config)# no router bgp 65100
```

Step-5

Assign S-1 an ASN 100 by using the command given below:

router bgp 100

```
sonic# configure
sonic(config)# router bgp 100
BGP is already running; AS is 65100
sonic(config)# no router bgp 65100
sonic(config)# router bgp 100
```

Step-6

Now configure S-2 so that it can form a connection with neighbors by using the following command given below:

neighbor <Interface address> remote-as <ASN>

```
sonic(config)# router bgp 100
sonic(config-router)# neighbor 192.168.10.3 remote-as 200
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-7

Before advertising network in BGP there is no entry of routes in routing table which can be seen by using the following command given below:

show ip bgp

```
sonic(config)# exit
sonic# show ip bgp
No BGP prefixes displayed, 0 exist
```

Step-8

Now configure S-2 for network advertisement in BGP by using the following command given below:

address-family ipv4 unicast

network <Network-ID>

exit-address-family

```
sonic(config)# router bgp 200
sonic(config-router)# address-family ipv4 unicast
sonic(config-router-af)# network 192.168.11.0/24
sonic(config-router-af)# network 192.168.10.0/24
sonic(config-router-af)# network 2.2.2.2/32
sonic(config-router-af)# exit-address-family
sonic(config-router)# exit
sonic(config)# exit
```

Step-9

Repeat steps 1-8 for the other switches.

Step-10

To check the status of BGP network advertisement, use the following command given below:

show ip bgp

```
sonic# show ip bgp
BGP table version is 3, local router ID is 2.2.2.2, vrf id 0
Default local pref 100, local AS 200
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop            Metric LocPrf Weight Path
*> 2.2.2.2/32       0.0.0.0                  0         32768 i
*> 192.168.10.0/24  0.0.0.0                  0         32768 i
*> 192.168.11.0/24  0.0.0.0                  0         32768 i

Displayed  3 routes and 3 total paths
sonic#
```

Result

Network advertisement: In IGPs (OSPF, EIGRP) "network command" enables a protocol on all of its interfaces. While in BGP, this command performs differently. It doesn't enable BGP on an interface rather it installs prefixes in its routing table.

eBGP Testing in SONiC

Introduction

This section explains the step-by-step procedure to test eBGP (routing protocol) and configure its features by running related commands in SONiC CLI.

Testbed Setup

Please refer to Testbed section

Network Topology

After importing images, now draw network topology in GNS3 using SONiC switches.

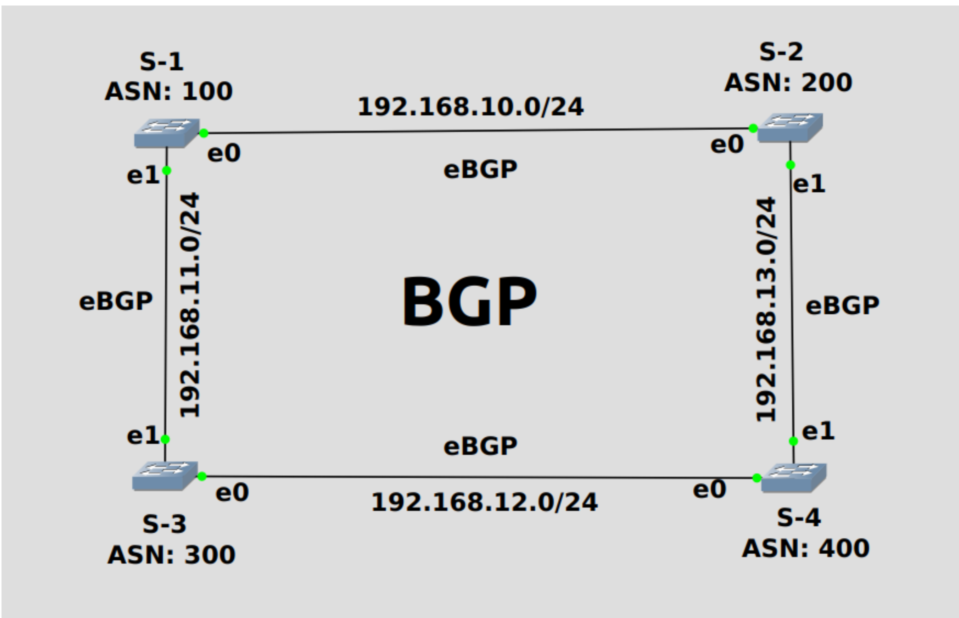


Fig: Network Topology

For the topology shown above, four switches (S-1, S-2, S-3 & S-4) are used.

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all hosts and switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A    fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A    fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A    fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A   fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A   fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A   fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A   fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A   fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A   fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A   fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A   fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A   fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A   fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A   fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A   fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A   fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A   fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below:

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet63

2nd Method

In this method, interface status can be changed by configuring "config_db" and the path is /etc/sonic. To configure "config_db", command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using “sudo config reload” command.

By following the above commands, make changes in "config_db” and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0",
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0",
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
-- INSERT --
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1^{st} command. So, change the status by using config_db.

Step-3

By default, all interfaces are routed (L3) and IP is assigned to them which can be seen using the command given below:

show ip interfaces

```
admin@sonic:~$ show ip interfaces
Interface    Master    IPv4 address/mask    Admin/Oper
-----------  --------  -------------------  ----------
Ethernet0              10.0.0.0/31          up/up
Ethernet4              10.0.0.2/31          up/up
Ethernet8              10.0.0.4/31          up/up
```

Remove the default IPs and assign those which are to be used in network topology. For this, the command is given below:

sudo config interface ip remove/add <interface_name> <ip_addr>

- admin@sonic:~$ sudo config interface ip remove Ethernet63 10.11.12.13/24

Step-4

By default, the BGP is in running mode with Autonomous System Number (ASN) 65100. First, stop this BGP process using the commands given below:

vtysh

configure

no router bgp 65100

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router bgp 100
BGP is already running; AS is 65100
sonic(config)# no router bgp 65100
```

Step-5

Assign S-1 an ASN 100 by using the command given below:

router bgp 100

```
sonic# configure
sonic(config)# router bgp 100
BGP is already running; AS is 65100
sonic(config)# no router bgp 65100
sonic(config)# router bgp 100
```

Step-6

Now configure S-1 so that it can form a connection with neighbors by using the following command given below:

neighbor <Interface address> remote-as <ASN>

```
sonic(config)# router bgp 100
sonic(config-router)# neighbor 192.168.10.3 remote-as 200
sonic(config-router)# neighbor 192.168.11.3 remote-as 300
sonic(config-router)# exit
sonic(config)# exit
sonic# write
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-7

Repeat steps 1-6 for the other switches.

Step-8

To check the bgp summary, use the following command given below:

show ip bgp summary

```
sonic# show ip bgp summary

IPv4 Unicast Summary (VRF default):
BGP router identifier 192.168.11.2, local AS number 100 vrf-id 0
BGP table version 0
RIB entries 0, using 0 bytes of memory
Peers 2, using 1447 KiB of memory

Neighbor        V         AS   MsgRcvd   MsgSent   TblVer  InQ OutQ  Up
192.168.10.3    4        200        19        20        0    0    0 00:
192.168.11.3    4        300        16        18        0    0    0 00:

Total number of neighbors 2
```

Step-9

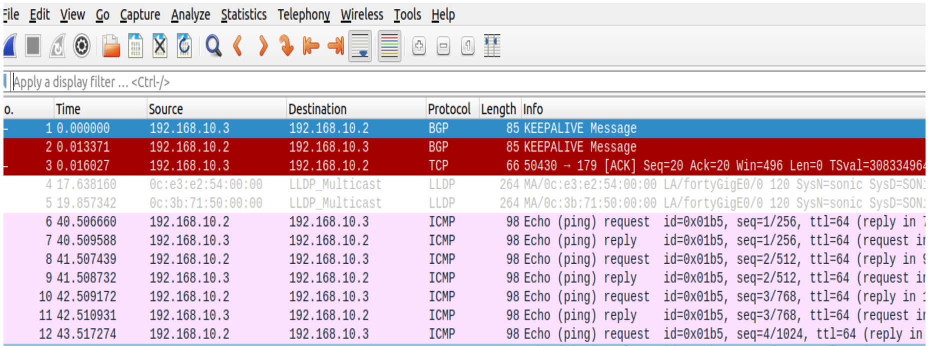
To check the status of bgp neighbors, use the following command given below:

show bgp neighbors

```
sonic# show bgp neighbors
BGP neighbor is 192.168.10.3, remote AS 200, local AS 100, external link
Hostname: sonic
  BGP version 4, remote router ID 192.168.13.2, local router ID 192.168.11.2
  BGP state = Established, up for 00:17:05
  Last read 00:00:05, Last write 00:00:05
  Hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    Extended Message: advertised and received
    AddPath:
      IPv4 Unicast: RX advertised and received
    Long-lived Graceful Restart: advertised and received
      Address families by peer:
    Route refresh: advertised and received(old & new)
    Enhanced Route Refresh: advertised and received
```

Step-10

For analysis of BGP, Wireshark has been used whose result is shown below:



Result

When ping request is generated from S-1 to S-2 and S-1 to S-3, a successful reply shows that BGP (eBGP) is successfully configured in the topology.

```
sonic# ping 192.168.10.3
PING 192.168.10.3 (192.168.10.3) 56(84) bytes of data.
64 bytes from 192.168.10.3: icmp_seq=1 ttl=64 time=6.38 ms
64 bytes from 192.168.10.3: icmp_seq=2 ttl=64 time=4.40 ms
64 bytes from 192.168.10.3: icmp_seq=3 ttl=64 time=6.67 ms
64 bytes from 192.168.10.3: icmp_seq=4 ttl=64 time=4.83 ms
64 bytes from 192.168.10.3: icmp_seq=5 ttl=64 time=4.13 ms
64 bytes from 192.168.10.3: icmp_seq=6 ttl=64 time=4.49 ms
64 bytes from 192.168.10.3: icmp_seq=7 ttl=64 time=4.29 ms
^C
--- 192.168.10.3 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 13ms
rtt min/avg/max/mdev = 4.131/5.026/6.670/0.970 ms
sonic# ping 192.168.11.3
PING 192.168.11.3 (192.168.11.3) 56(84) bytes of data.
64 bytes from 192.168.11.3: icmp_seq=1 ttl=64 time=5.13 ms
64 bytes from 192.168.11.3: icmp_seq=2 ttl=64 time=3.91 ms
64 bytes from 192.168.11.3: icmp_seq=3 ttl=64 time=4.10 ms
64 bytes from 192.168.11.3: icmp_seq=4 ttl=64 time=4.30 ms
64 bytes from 192.168.11.3: icmp_seq=5 ttl=64 time=4.21 ms
```

Layer 3 (IPv6)

RIPng (IPv6)

RIPng Overview

The Routing Information Protocol next generation (RIPng) is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng exchanges routing information used to compute routes and is intended for IP version 6 (IPv6)-based networks. Here are some of its features:

- Just like RIP for IPv4, it uses hop count as the metric.
- sends updates every 30 seconds.
- RIPng messages use the UDP port 521 and the multicast address of FF02::9

Testbed Setup

Please refer to Testbed section

Network Topology

Now draw network topology in GNS3 using SONiC switches. In our case, four switches (S-1, S-2, S-3 & S-4) are used. Between S-1 and S-2, S-1 and S-3, S-2 and S-4, S-3 and S-4, global unicast network addresses 2001::/64, 2002::/64, 2004::/64 and 2003::/64 have been used respectively.

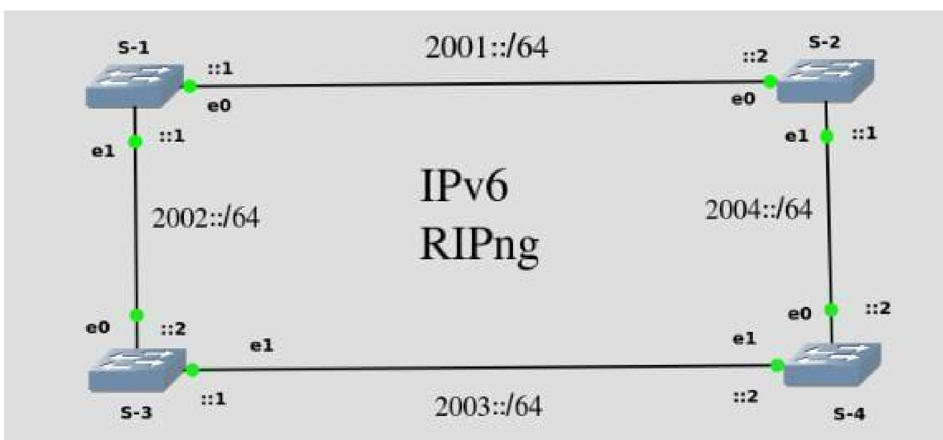


Fig: Network Topology

Note: Be patient while using SONiC CLI because it takes some time while show results.

Configurations

For the above topology, all switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given [here]

To configure switch S-1, use the following steps given below:

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A     fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A     fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A     fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A    fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A    fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A    fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A    fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A    fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A    fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A    fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A    fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A    fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A    fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A    fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A    fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A    fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A    fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operational "down" but administratively "up". In most cases, operational status is usually down but sometimes it is "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet4

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, the command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using “sudo config reload” command.

By following the above commands, make changes in "config_db” and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces have been assigned Link-Local address (IPv6) which can be seen by using the command given below:

show ipv6 interfaces

```
admin@sonic:~$ show ipv6 interfaces
Interface    Master    IPv6 address/mask                     Admin/Oper    BGP Neighbor    Neighbor IP
-----------  --------  ------------------------------------  ------------  --------------  -------------
Bridge                 fe80::10dd:90ff:fe49:f40f%Bridge/64   up/down       N/A             N/A
Ethernet0              fe80::ec1:42ff:fe5f:0%Ethernet0/64    up/up         N/A             N/A
Ethernet4              fe80::ec1:42ff:fe5f:0%Ethernet4/64    up/up         N/A             N/A
Ethernet8              fe80::ec1:42ff:fe5f:0%Ethernet8/64    up/up         N/A             N/A
Ethernet12             fe80::ec1:42ff:fe5f:0%Ethernet12/64   up/up         N/A             N/A
```

To assign Global Unicast address to interfaces, use the following commands given below:

vtysh

configure

interface <interface_name>

ip address <ipv6 address>

VTYSH is a shell for FRR daemons. It amalgamates all the CLI commands defined in each of the daemons and presents them to the user in a single shell, which saves the user from having to telnet each of the daemons and use their individual shells.

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# interface Ethernet0
sonic(config-if)# ip address 2001::1/64
sonic(config-if)# interface Ethernet4
sonic(config-if)# ip address 2002::1/64
sonic(config-if)# exit
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/isisd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is better to practice saving configurations after executing two or three commands.

Step-4

BGP container: Runs the supported routing stack which is FRR. Even though the container is named after the routing protocol being used (bgp), in reality, these routing stacks can run various other protocols (such as ripng, OSPF, isis, LDP, etc).

By default, the ripng daemon is not running. For ripngd to be in the running mode use the following commands given below:

docker exec -it bgp bash
cd usr/lib/frr
ls
./ripngd &
exit

The **"docker exec -it bgp bash" command** is used to start an interactive shell session inside a Docker container named "bgp".

The **"./ripngd &" command** is used to start the RIPng daemon in the background on a Unix or Linux system. The "&" symbol at the end of the command is a shell operator that tells the system to execute the command in the background, allowing the user to continue to use the terminal for other commands.

```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd /usr/lib/frr
root@sonic:/usr/lib/frr# ls
babeld  eigrpd      frrcommon.sh  ldpd    ospfd  pimd    staticd   watchfrr.sh
bfdd    fabricd     frrinit.sh    nhrpd   pathd  ripd    vrrpd     zebra
bgpd    frr-reload  isisd         ospf6d  pbrd   ripngd  watchfrr
root@sonic:/usr/lib/frr# ./ripngd &
[1] 8539
root@sonic:/usr/lib/frr# exit
exit
```

Step-5

RIPng must be enabled before carrying out any of the ripng commands by using the commands given below:

vtysh
configure
router ripng

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router ripng
```

Step-6

After enabling RIPng, assign all the network addresses to which the switch is directly connected by using the command given below:

network <network address>

```
sonic(config-router)# network 2001::/64
sonic(config-router)# network 2002::/64
sonic(config-router)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is highly recommended to save the configurations using the “write” command.

Step-7

Repeat steps 1-6 for other switches as well.

Step-8

Check RIPng status using the commands given below:

show ip ripng

show ipv6 route

```
sonic# show ip ripng
Codes: R - RIPng, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface, (a/S) - aggregated/Suppressed

   Network      Next Hop                      Via     Metric Tag Time
C(i) 2001::/64
                  ::                          self       1    0
C(i) 2002::/64
                  ::                          self       1    0
R(n) 2003::/64
                  fe80::ef6:c8ff:fe8a:0       Ethernet4  2    0  02:57
R(n) 2004::/64
                  fe80::e3d:20ff:fe33:0       Ethernet0  2    0  02:57
```

```
admin@sonic:~$ show ipv6 route
Codes: K - kernel route, C - connected, S - static, R - RIPng,
       O - OSPFv3, I - IS-IS, B - BGP, N - NHRP, T - Table,
       v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

C>* 2001::/64 is directly connected, Ethernet0, 00:23:19
C>* 2002::/64 is directly connected, Ethernet4, 00:23:04
R>* 2003::/64 [120/2] via fe80::ef6:c8ff:fe8a:0, Ethernet4, weight 1, 00:11:32
R>* 2004::/64 [120/2] via fe80::e3d:20ff:fe33:0, Ethernet0, weight 1, 00:12:47
```

To check the configuration file of RIPng, use the following commands given below:

docker exec -it bgp bash

cd /etc/frr

ls

cat ripngd.conf

```
root@sonic:/etc/frr# cat ripngd.conf
!
! Zebra configuration saved from vty
!   2023/03/08 07:15:56
!
frr version 8.2.2
frr defaults traditional
!
hostname sonic
!
!
!
!
!
router ripng
 network 2001::/64
 network 2002::/64
exit
!
!
!
!
root@sonic:/etc/frr#
```

Result

After configuring the switches as per topology, it is clearly seen that switch S-1 can send traffic to S-4 and other switches because, In Linux, the default TTL value is 64. Each router that the packet passes through decrements the TTL value by one. By looking at the figure which is given below, TTL has been decremented to 63 because the packet reaches S-4 by passing through only one router. So, RIPng is successfully configured in the topology.

```
sonic# ping 2003::2
PING 2003::2(2003::2) 56 data bytes
64 bytes from 2003::2: icmp_seq=1 ttl=63 time=19.8 ms
64 bytes from 2003::2: icmp_seq=2 ttl=63 time=7.82 ms
64 bytes from 2003::2: icmp_seq=3 ttl=63 time=8.57 ms
64 bytes from 2003::2: icmp_seq=4 ttl=63 time=7.58 ms
64 bytes from 2003::2: icmp_seq=5 ttl=63 time=8.39 ms
^C
```

IS-IS (IPv6)

Introduction

The IS-IS protocol is an interior gateway protocol (IGP) that uses link-state information to make routing decisions.

IS-IS is a link-state IGP that uses the shortest-path-first (SPF) algorithm to determine routes. IS-IS evaluates the topology changes and determines whether to perform a full SPF recalculation or a partial route calculation (PRC). This protocol originally was developed for routing International Organization for Standardization (ISO) Connectionless Network Protocol (CLNP) packets.

Like OSPF routing, IS-IS uses hello packets that allow network convergence to occur quickly when network changes are detected. IS-IS uses the SPF algorithm to determine routes. Using SPF, IS-IS evaluates network topology changes and determines if a full or partial route calculation is required.

NOTE**: Because IS-IS uses ISO addresses, the configuration of IP version 6 (IPv6) and IP version 4 (IPv4) implementations of IS-IS is identical**.

IS-IS Terminology

An IS-IS network is a single autonomous system (AS), also called a routing domain, that consists of end systems and intermediate systems. End systems are network entities that send and receive packets. Intermediate systems send and receive packets and relay (forward) packets. (Intermediate system is the Open System Interconnection [OSI] term for a router.) ISO packets are called network PDUs.

In IS-IS, a single AS can be divided into smaller groups called areas. Routing between areas is organized hierarchically, allowing a domain to be administratively divided into smaller areas. This organization is accomplished by configuring Level 1 and Level 2 intermediate systems. Level 1 systems route within an area; when the destination is outside an area, they route toward a Level 2 system. Level 2 intermediate systems route between areas and toward other ASs. No IS-IS area functions strictly as a backbone.

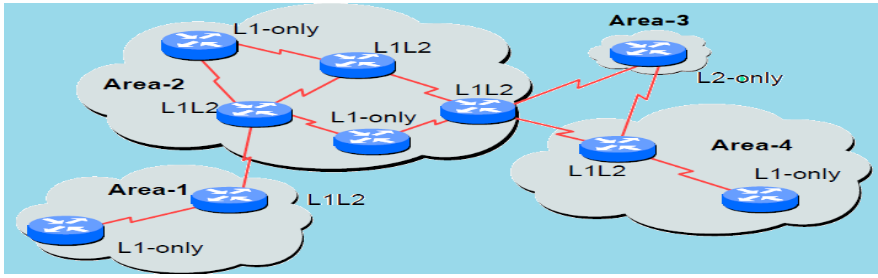


Figure: IS-IS Levels

Level 1 Routers—Level 1 routers share intra-area routing information.

Level 2 Routers—Level 2 routers share inter-area information about IP addresses available within each area.

Level 3 Routers—IS-IS routers can act as both Level 1 and Level 2 routers, sharing intra-area routes with other Level 1 router and interarea routes with other Level 2 routers.

The propagation of link-state updates is determined by the level boundaries. All routers within a level maintain a complete link-state database of all other routers in the same level. Each router then uses the Dijkstra algorithm to determine the shortest path from the local router to other routers in the link-state database.

ISO Network Addresses

IS-IS uses ISO network addresses. Each address identifies a point of connection to the network, such as a router interface, and is called a network service access point (NSAP).

An end system can have multiple NSAP addresses, in which case the addresses differ only by the last byte (called the n-selector). Each NSAP represents a service that is available at that node. In addition to having multiple services, a single node can belong to multiple areas.

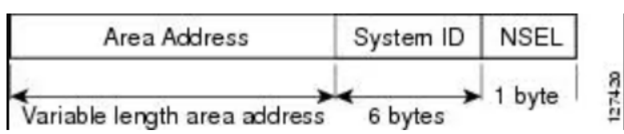


Figure: NET Address

Each network entity also has a special network address called a network entity title (NET). Structurally, an NET is identical to an NSAP address but has an n-selector of 00. Most end systems and intermediate systems have one NET. Intermediate systems that participate in multiple areas can have multiple NETs.

The following ISO addresses illustrate the IS-IS address format:
49.0001.00a0.c96b.c490.00
49.0001.2081.9716.9018.00

NETs take several forms, depending on your network requirements. NET addresses are hexadecimal and range from 8 octets to 20 octets in length. Generally, the format consists of an authority and format Identifier (AFI), a domain ID, an area ID, a system identifier, and a selector. The simplest format omits the domain ID and is 10 octets long. For example, the NET address 49.0001.1921.6800.1001.00 consists of the following parts:

- 49—AFI
- 0001—Area ID
- 1921.6800.1001—System identifier
- 00—Selector

The system identifier must be unique within the network. For an IP-only network, we recommend using the IP address of an interface on the router. Configuring a loopback NET address with the IP address is helpful when troubleshooting is required on the network.

The first portion of the address is the area number, which is a variable number from 1 through 13 bytes. The first byte of the area number (49) is the authority and format indicator (AFI). The next bytes are the assigned domain (area) identifier, which can be from 0 to 12 bytes. In the examples above, the area identifier is 0001.

The next six bytes form the system identifier. The system identifier can be any six bytes that are unique throughout the entire domain. The system identifier commonly is the media access control (MAC) address (as in the first example, 00a0.c96b.c490) or the IP address expressed in binary-coded decimal (BCD) (as in the second example, 2081.9716.9018, which corresponds to IP address 208.197.169.18). The last byte (00) is the n-selector.

NOTE: **The system identifier cannot be 0000.0000.0000. All 0s is an illegal setting, and adjacency is not formed with this setting.**

IS-IS PDU Types

ISs exchange routing information with their peers using protocol data units (PDUs). The following types of PDUs are used:

- IIHs
- LSPs
- SNPs

IIHs

Intermediate System-to-Intermediate System Hello PDUs (IIHs) are exchanged between IS neighbours on circuits on which the IS-IS protocol is enabled. IIHs include the system ID of the sender, the assigned area address(es), and the identity of neighbours on that circuit that are known to the sending IS. Additional optional information may also be included.

There are three types of IIHs:

Point-to-Point IIHs—These are sent on point-to-point circuits.

Level-1 LAN IIHs—These are sent on multiaccess circuits when the sending IS operates as a Level-1 device on that circuit.

Level-2 LAN IIHs—These are sent on multiaccess circuits when the sending IS operates as a Level-2 device on that circuit.

LSPs

An IS generates Link-State PDUs (LSPs) to advertise its neighbours and the destination that are directly connected to the IS. An LSP is uniquely identified by the following:

System ID of the IS that generated the LSP

pseudonode ID—This value is always 0 except when the LSP is a pseudonode LSP (see "Operation of IS-IS on Multiaccess Circuits" section.

LSP number (0 to 255)

32-bit sequence number

Whenever a new version of an LSP is generated, the sequence number is incremented.

Level-1 LSPs are generated by ISs that support Level 1. The Level-1 LSPs are flooded throughout the Level-1 area. The set of Level-1 LSPs generated by all Level-1 ISs in an area is the Level-1 LSP Database (LSPDB). All Level-1 ISs in an area will have an identical Level-1 LSPDB and will therefore have an identical network connectivity map for the area.

Level-2 LSPs are generated by ISs that support Level 2. Level-2 LSPs are flooded throughout the Level-2 subdomain. The set of Level-2 LSPs generated by all Level-2 ISs in the domain is the Level-2 LSP Database (LSPDB). All Level-2 ISs will have an identical Level-2 LSPDB and will therefore have an identical connectivity map for the Level-2 subdomain.

SNPs

Sequence Number PDUs (SNPs) contain a summary description of one or more LSPs. There are two types of SNPs for both Level 1 and Level 2:

Complete Sequence Number PDUs (CSNPs) are used to send a summary of the LSPDB that an IS has for a given level.

Partial Sequence Number PDUs (PSNPs) are used to send a summary of a subset of the LSPs for a given level that an IS either has in its database or needs to obtain.

Testbed Setup

Please refer to Testbed section

Network Topology

Draw network topology in GNS3 using SONiC switches. In our case, four switches (S-1, S-2, S-3 & S-4) are used. Between S-1 and S-2, S-1 and S-3, S-2 and S-4, S-3 and S-4, global unicast network addresses 2001::/64, 2002::/64, 2004::/64 and 2003::/64 have been used respectively.

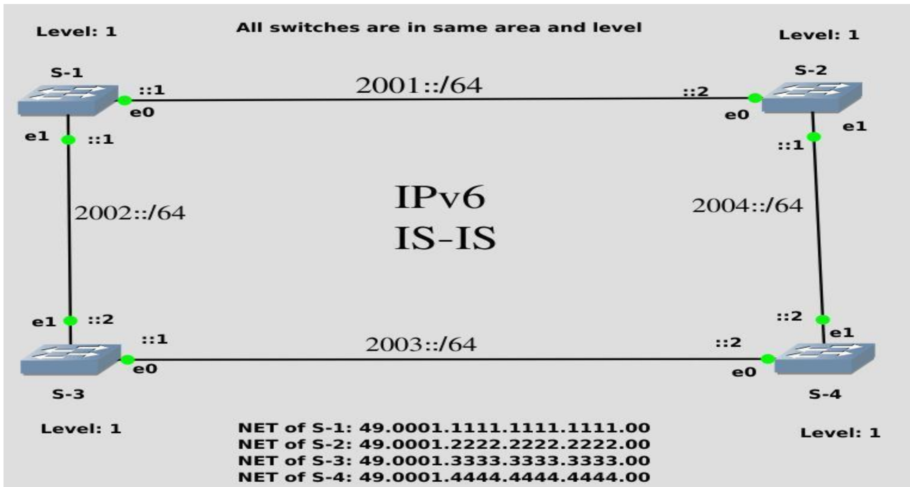


Fig: Network Topology

Note: Be patient while using SONiC CLI because it takes some time while showing results.

Configurations

For the above topology, all switches are first configured before sending traffic. First, the switch (S-1) is configured, and the same steps are repeated for the other switches. Command Reference guide is also available on GitHub for SONiC, whose link is given here

Follow these steps to configure Switch-1.

Step-1

Check the status of interfaces by using the command:

show interfaces status

```
admin@sonic:~$ show interfaces status
  Interface            Lanes    Speed    MTU    FEC           Alias    Vlan    Oper    Admin    Type    Asym PFC
-----------  ---------------  -------  -----  -----  --------------  ------  ------  -------  ------  ----------
  Ethernet0      25,26,27,28      40G   9100    N/A     fortyGigE0/0  routed    down       up     N/A         N/A
  Ethernet4      29,30,31,32      40G   9100    N/A     fortyGigE0/4  routed    down       up     N/A         N/A
  Ethernet8      33,34,35,36      40G   9100    N/A     fortyGigE0/8  routed    down       up     N/A         N/A
 Ethernet12      37,38,39,40      40G   9100    N/A    fortyGigE0/12  routed    down       up     N/A         N/A
 Ethernet16      45,46,47,48      40G   9100    N/A    fortyGigE0/16  routed    down       up     N/A         N/A
 Ethernet20      41,42,43,44      40G   9100    N/A    fortyGigE0/20  routed    down       up     N/A         N/A
 Ethernet24          1,2,3,4      40G   9100    N/A    fortyGigE0/24  routed    down       up     N/A         N/A
 Ethernet28          5,6,7,8      40G   9100    N/A    fortyGigE0/28  routed    down       up     N/A         N/A
 Ethernet32      13,14,15,16      40G   9100    N/A    fortyGigE0/32  routed    down       up     N/A         N/A
 Ethernet36        9,10,11,12     40G   9100    N/A    fortyGigE0/36  routed    down       up     N/A         N/A
 Ethernet40      17,18,19,20      40G   9100    N/A    fortyGigE0/40  routed    down       up     N/A         N/A
 Ethernet44      21,22,23,24      40G   9100    N/A    fortyGigE0/44  routed    down       up     N/A         N/A
 Ethernet48      53,54,55,56      40G   9100    N/A    fortyGigE0/48  routed    down       up     N/A         N/A
 Ethernet52      49,50,51,52      40G   9100    N/A    fortyGigE0/52  routed    down       up     N/A         N/A
 Ethernet56      57,58,59,60      40G   9100    N/A    fortyGigE0/56  routed    down       up     N/A         N/A
 Ethernet60      61,62,63,64      40G   9100    N/A    fortyGigE0/60  routed    down       up     N/A         N/A
 Ethernet64      69,70,71,72      40G   9100    N/A    fortyGigE0/64  routed    down       up     N/A         N/A
```

- In the above figure, all interfaces are operationally "down" but administratively "up". In most cases, operational status is usually down but sometimes "up" after running devices in GNS3.

Step-2

There are two methods to change operational status, which are given below

1st Method

To "up" the operational status of an interface, use the following command:

- sudo config interface startup <interface_name> (for 201904+ version)
- admin@sonic:~$ sudo config interface startup Ethernet4

2nd Method

In this method, interface status can be changed by configuring “config_db” and the path is /etc/sonic. To configure "config_db”, the command is given below:

sudo vi config_db.json

Note: It is highly recommended that first save all the configurations and then reload it using "sudo config reload" command.

By following the above commands, make changes in "config_db" and then save it.

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "mtu": "9100"
```

```
},
"PORT": {
    "Ethernet0": {
        "lanes": "25,26,27,28",
        "alias": "fortyGigE0/0"
        "index": "0",
        "speed": "40000",
        "admin_status": "up",
        "oper_status": "up",
        "mtu": "9100"
    },
    "Ethernet4": {
        "lanes": "29,30,31,32",
        "alias": "fortyGigE0/4"
```

Note: Using the first method to change interface status is recommended. Sometimes interface status remains down after using 1st command. So, change the status by using config_db.

Step-3

By default, all interfaces have been assigned Link-Local address (IPv6) which can be seen by using the command given below:

show ipv6 interfaces

```
admin@sonic:~$ show ipv6 interfaces
Interface    Master    IPv6 address/mask                     Admin/Oper    BGP Neighbor    Neighbor IP
-----------  --------  ------------------------------------  ------------  --------------  -------------
Bridge                 fe80::10dd:90ff:fe49:f40f%Bridge/64    up/down       N/A             N/A
Ethernet0              fe80::ec1:42ff:fe5f:0%Ethernet0/64     up/up         N/A             N/A
Ethernet4              fe80::ec1:42ff:fe5f:0%Ethernet4/64     up/up         N/A             N/A
Ethernet8              fe80::ec1:42ff:fe5f:0%Ethernet8/64     up/up         N/A             N/A
Ethernet12             fe80::ec1:42ff:fe5f:0%Ethernet12/64    up/up         N/A             N/A
```

To assign Global Unicast address to interfaces, use the following commands given below:

vtysh

configure

interface <interface_name>

ip address <ipv6 address>

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# interface Ethernet0
sonic(config-if)# ip address 2001::1/64
sonic(config-if)# interface Ethernet4
sonic(config-if)# ip address 2002::1/64
sonic(config-if)# exit
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/isisd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is better practice to save configurations after executing two or three commands.

Step-4

By default, the IS-IS daemon is not running. For isisd to be in the running mode use the following commands given below:

docker exec -it bgp bash

cd usr/lib/frr

ls

./isisd &

Exit

The **"docker exec -it bgp bash" command** is used to start an interactive shell session inside a Docker container named "bgp".

The **"./isisd &" command** is used to start the IS-IS daemon in the background on a Unix or Linux system. The "&" symbol at the end of the command is a shell operator that tells the system to execute the command in the background, allowing the user to continue to use the terminal for other commands.

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router isis
% Command incomplete: router isis
sonic(config)# router isis humza
isisd is not running
```

```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd /usr/lib/frr
root@sonic:/usr/lib/frr# ls
babeld  eigrpd      frrcommon.sh  ldpd    ospfd  pimd    staticd   watchfrr.sh
bfdd    fabricd     frrinit.sh    nhrpd   pathd  ripd    vrrpd     zebra
bgpd    frr-reload  isisd         ospf6d  pbrd   ripngd  watchfrr
root@sonic:/usr/lib/frr# ./isisd &
[1] 136
```

Step-5

IS-IS must be enabled before carrying out any of the IS-IS commands by using the commands given below:

vtysh
configure
router isis WORD [vrf NAME]

The command "**router isis WORD [vrf NAME]"** enables or disables the ISIS process by specifying the ISIS domain with 'WORD'. The "**WORD"** may be any number, alphabet or word.

```
admin@sonic:~$ vtysh

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

sonic# configure
sonic(config)# router isis
% Command incomplete: router isis
sonic(config)# router isis humza
```

Step-6

After enabling IS-IS, assign the net address and level to the router by using the commands given below:

net <net-address>
is-type [level-1 | level-1-2 | level-2-only]

```
sonic(config)# router isis humza
sonic(config-router)# net 49.0001.1111.1111.1111.00
sonic(config-router)# is-type level-1
sonic(config-router)# exit
sonic(config)# exit
```

```
sonic(config)# exit
sonic# write
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Configuration saved to /etc/frr/zebra.conf
Configuration saved to /etc/frr/ospfd.conf
Configuration saved to /etc/frr/bgpd.conf
Configuration saved to /etc/frr/staticd.conf
```

Note: It is highly recommended to save the configurations using the "write" command.

Step-7

First activate IS-IS on the interface and then configure the circuit type for that interface by using the commands given below:

<ipv6> router isis WORD

The above command activates ISIS adjacency on the interface. Note that the name of ISIS instance must be the same as the one used to configure the ISIS process.

isis circuit-type [level-1 | level-1-2 | level-2]

```
sonic# configure
sonic(config)# interface Ethernet0
sonic(config-if)# ipv6 router isis humza
sonic(config-if)# isis circuit-type level-1
```

Step-8

Repeat steps 1-7 for other switches as well.

Step-9

Check IS-IS status using the commands given below:

show isis neighbor

show isis route

show isis summary

```
sonic# sh isis neighbor
Area humza:
  System Id           Interface   L  State        Holdtime SNPA
 sonic                Ethernet0   1  Up           29       0c70.6fb8.0000
 sonic                Ethernet4   1  Up           30       0cba.6032.0000
```

The screenshot above indicates that switch S-1 has two neighbors, demonstrating the successful implementation of our topology. This is evident because switch S-1 is connected directly to two switches using Ethernet0 and Ethernet4 interfaces.

```
sonic# sh isis route
Area humza:
IS-IS L1 IPv6 routing table:

 Prefix     Metric  Interface  Nexthop                  Label(s)
 ---------------------------------------------------------------
 2001::/64  0       -          -                        -
 2002::/64  0       -          -                        -
 2003::/64  20      Ethernet4  fe80::eba:60ff:fe32:0    -
 2004::/64  30      Ethernet4  fe80::eba:60ff:fe32:0    -
```

The above figure shows the routing table of S-1 which consists of four routes. Two of them “2001::/64 and 2002::/64” do not have Nexthop addresses because they are directly connected to switch S-1 while routes “2003::/64 and 2004::/64” have Link-local addresses (Nexthop) because they are not directly connected to switch

```
sonic# sh isis summary
vrf                 : default
Process Id          : 508
System Id           : 1111.1111.1111
Up time             : 01:05:37 ago
Number of areas : 1
Area humza:
  Net: 49.0001.1111.1111.1111.00
  TX counters per PDU type:
     L1 IIH: 1679
     L1 LSP: 31
    L1 CSNP: 528
   LSP RXMT: 0
  RX counters per PDU type:
     L1 IIH: 1678
     L1 LSP: 13
    L1 PSNP: 1
  Level-1:
    LSP0 regenerated: 9
         LSPs purged: 0
    SPF:
      minimum interval   : 1
    IPv6 route computation:
      last run elapsed   : 00:06:51 ago
      last run duration : 835 usec
      run count          : 25
```

To check the configuration file of IS-IS, use the following commands given below:

docker exec -it bgp bash

cd /etc/frr

ls

cat isisd.conf

```
admin@sonic:~$ docker exec -it bgp bash
root@sonic:/# cd /etc/frr/
root@sonic:/etc/frr# ls
bgpd.conf      isisd.conf      staticd.conf      vtysh.conf  zebra.co
bgpd.conf.sav  isisd.conf.sav  staticd.conf.sav  zebra.conf
root@sonic:/etc/frr# cat isisd.conf
```

```
!
! Zebra configuration saved from vty
!   2023/03/17 13:05:18
!
frr version 8.2.2
frr defaults traditional
!
hostname sonic
!
!
!
!
interface Ethernet0
 ipv6 router isis humza
 isis circuit-type level-1
exit
!
interface Ethernet4
 ipv6 router isis humza
 isis circuit-type level-1
exit
!
!
"isisd.conf" 31L, 370C
```

Result

The configuration of IS-IS (Intermediate System to Intermediate System) routing protocol with switches S-1, S-2, S-3, and S-4 and routes 2001::/64, 2002::/64, 2003::/64, and 2004::/64 is critical for efficient and reliable transfer of data packets across the network. To ensure proper configuration, administrators should define network entity titles (NET), set up areas and levels, and enable appropriate routing protocols. Additionally, implementing ping and Time-to-Live (TTL) decrement can help verify the proper configuration of IS-IS by allowing administrators to test network connectivity and ensure that packets are being forwarded correctly. The default TTL value in Linux is 64, and as a packet travels through each router, the TTL value is reduced by one. During the generation of ping for interface 2001::2/64, the TTL did not decrement, remaining at 64. However, for interfaces 2003::2/64 and 2004::2/64, the TTL value was decremented by one to 63, indicating the successful implementation of our topology.

```
admin@sonic:~$ ping 2003::2
PING 2003::2(2003::2) 56 data bytes
64 bytes from 2003::2: icmp_seq=1 ttl=63 time=22.4 ms
64 bytes from 2003::2: icmp_seq=2 ttl=63 time=8.18 ms
64 bytes from 2003::2: icmp_seq=3 ttl=63 time=9.03 ms
^C
--- 2003::2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 8.180/13.213/22.436/6.530 ms
admin@sonic:~$ ping 2001::2
PING 2001::2(2001::2) 56 data bytes
64 bytes from 2001::2: icmp_seq=1 ttl=64 time=7.56 ms
64 bytes from 2001::2: icmp_seq=2 ttl=64 time=4.38 ms
64 bytes from 2001::2: icmp_seq=3 ttl=64 time=4.67 ms
^C
--- 2001::2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 4.378/5.535/7.561/1.437 ms
admin@sonic:~$ ping 2004::2
PING 2004::2(2004::2) 56 data bytes
64 bytes from 2004::2: icmp_seq=1 ttl=63 time=7.62 ms
64 bytes from 2004::2: icmp_seq=2 ttl=63 time=8.20 ms
64 bytes from 2004::2: icmp_seq=3 ttl=63 time=8.27 ms
64 bytes from 2004::2: icmp_seq=4 ttl=63 time=8.50 ms
^C
--- 2004::2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 7.618/8.146/8.497/0.324 ms
admin@sonic:~$
```

Conclusion

This document is a living resource, and our commitment to providing up-to-date information extends beyond its current scope. In the upcoming iterations, we plan to expand the coverage of SONiC within GNS3 by testing additional features such as VxLAN, MLAG (Multi-Chassis Link Aggregation), VRRP (Virtual Router Redundancy Protocol), and more. By incorporating these advanced functionalities, we aim to offer readers a comprehensive understanding of SONiC's capabilities and its potential applications in various network scenarios. Stay tuned for future updates as we continue to explore and showcase the power and versatility of SONiC within the GNS3 environment.

References

- https://study-ccna.com/what-is-a-vlan/
- https://support.huawei.com/enterprise/en/doc/EDOC1000142081/9c0373e/intra-vlan-communication
- https://study-ccna.com/access-and-trunk-ports/
- *FRRUserManual* https://docs.frrouting.org/_/downloads/en/latest/pdf/
- https://community.cisco.com/t5/routing/is-is-nsap-address/td-p/417266
- https://www.cisco.com/c/en/us/support/docs/ip/integrated-intermediate-system-to-intermediate-system-is-is/13795-is-is-ip-config.html
- https://www.juniper.net/documentation/us/en/software/junos/is-is/topics/concept/is-is-routing-overview.html
- https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_isis/configuration/15-mt/irs-15-mt-book/irs-ovrw-cf.html
- https://techhub.hpe.com/eginfolib/networking/docs/switches/5700/5998-5589r_l3-ip-rtng_cg/content/446940293.htm