8

Preparing for Certification Using GNS3

This is the chapter where you get to do the work. No matter what level of certification you are going for, these are the tips and exercises that will be useful for you.

The following topics will be covered in this chapter:

- Exercises to prepare you for CCENT/CCNA:
    - Two-router tango
    - Simple switching
    - VLAN variety
    - OSPF operation
    - EIGRP excitement
- Exercises to boost/assist/advance you to CCNP-level certifications:
    - EIGRP
    - Multi-area OSPF
    - eBGP
- Striving for the "Holy Grail"—how to use GNS3 to prepare for the CCIE R&S lab
    - Sample CCIE practice lab design using simulated switches
    - Sample CCIE practice lab design using your own switches

These exercises are aimed at specific objectives with the Cisco certification exams in mind, but could easily be adapted for Junos or Vyatta. Even if you are not studying for certifications, reading this chapter will give you tips on how to set up a lab environment for any simulation.

This chapter has been written to cater to a variety of router technology skill levels, but the more advanced exercises require more advanced GNS3 familiarity as well. So even if you are using GNS3 to study for advanced certifications, you may still find it useful to at least set up some of the earlier topologies to gain familiarity.

By the end of this chapter, you will have a complex simulation environment, ready to tackle some extremely diverse simulations.

Getting ready for CCENT/CCNA

Here are some practical exercises to test your mettle and to see if you meet the exacting standards set for the Cisco ICND1 and ICND2 exams. The first exercise is to ensure you are comfortable with the GNS3 environment and can do the most basic of all exercises—get two routers routing!

Two router tango

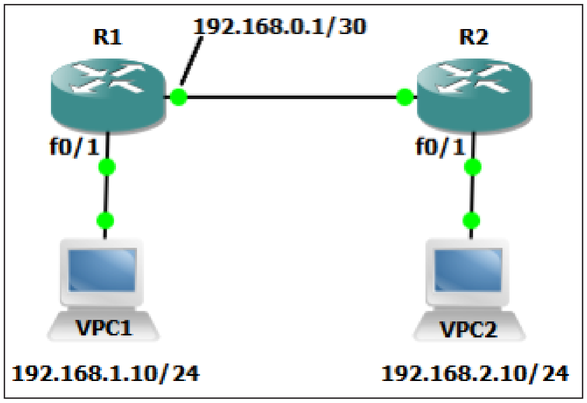
Build a topology with two Cisco routers, each with an attached Virtual PC. Ideally, you should be able to design your own IP addressing scheme as well, but use the following diagram if you can't manage your own design.

Principle objective

The two virtual PCs should be able to ping each other.

Topology

Use the partial IP addressing scheme shown as follows (you fill in the gaps), or better still, design your own IP addressing scheme. Some information, such as the default gateway IP of the Virtual PCs, has been deliberately omitted because that will depend on how you choose to complete the design.



Validation

From the VPCS, the following commands must produce exactly the same output (apart from response times, which are semi-random, and the IP addresses assigned to f0/1 of each router):

```
VPCS[1]> ping 192.168.0.1 -c 2
192.168.0.1 icmp_seq=1 ttl=255 time=30.992 ms
192.168.0.1 icmp_seq=2 ttl=255 time=15.475 ms

VPCS[1]> ping 192.168.0.7 -c 2
*192.168.1.1 icmp_seq=1 ttl=255 time=47.242 ms
  (ICMP type:3, code:1, Destination host unreachable)
*192.168.1.1 icmp_seq=2 ttl=255 time=15.393 ms
  (ICMP type:3, code:1, Destination host unreachable)

VPCS[1]> ping 192.168.2.7 -c 2
192.168.2.7 icmp_seq=1 timeout
192.168.2.7 icmp_seq=2 timeout

VPCS[1]> 2
VPCS[2]> trace 192.168.1.10
trace to 192.168.1.10, 8 hops max, press Ctrl+C to stop
 1   192.168.2.1   14.774 ms  15.654 ms  30.584 ms
 2   192.168.0.1   62.937 ms  30.513 ms  31.996 ms
 3   *192.168.1.10   77.572 ms
  (ICMP type:3, code:3, Destination port unreachable)
```

Hints

The exercise does not specify any routing criteria. You *will* need to use either static routes or OSPF routing to make the exercise work. Make sure you can perform both methods.

Variations

You might like to try the following variations:

- You must work with the IP address allocation of `192.168.15.64/26` and ensure that there are sufficient IP addresses to cater to 20 hosts on the VPC1 subnet.
- Assign all IP addresses from the range `fc00:0000:0000:abc0::/62`. Use OSPFv3 to ensure connectivity.

Simple switching

It is possible to use the NM-16ESW module in several routers to provide some fundamental switching functions. GNS3 has the following two features that help you use the NM-16ESW module to create a Cisco layer 2 switch:

- A device called the EtherSwitch router in the Switches devices toolbar
- A default `baseconfig_sw.txt` obj type reqd, mate suitable for the EtherSwitch router

When you add the EtherSwitch router device to your topology, GNS3 looks to see if you have a Cisco c3700 platform image configured in the **IOS images and hypervisors** option under the **Edit** menu. If so, it puts a 3700 router loaded with a NM-16ESW (a layer 2 switch module) right into your topology.

GNS3 also looks to see if the file `baseconfig_sw.txt` exists, and if so, uses it as the startup configuration file rather than the file you may have assigned as the **Base config** file for this image.

For this exercise, you will need to have a 37xx image available. I used a `c3725-adventerprisek9-mz.124-15.T10.bin` image.

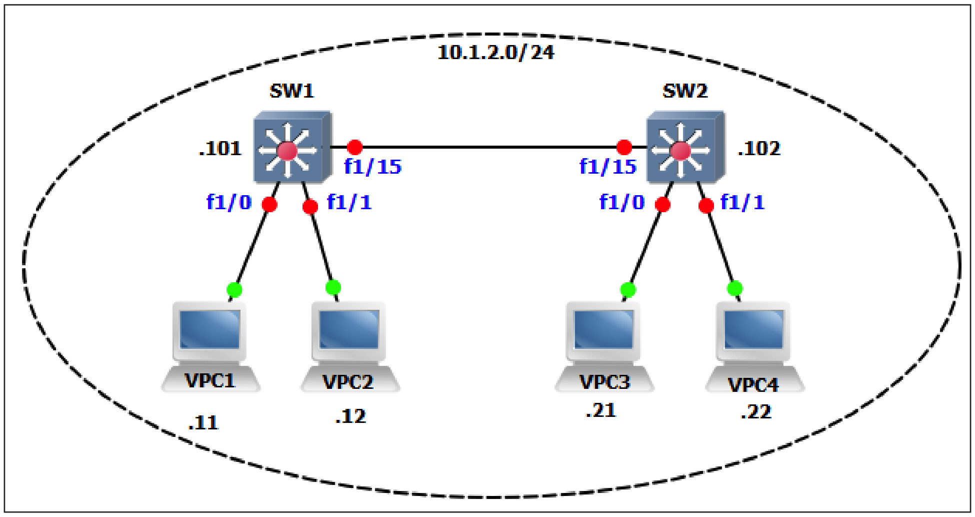
Principle objectives

To attain familiarity with basic switching concepts and, specifically, the operation of the following Cisco switches (from the 100-101 ICND1 Exam Topics available at `http://www.cisco.com/web/learning/exams/list/icnd1b.html#~Topics`).

- Broadcast Domains
- CAM Table

Topology

Build the topology, shown as follows, using EtherSwitch routers and VPCs. The IP address will be assigned during the course of the exercise. Pay particular attention to the interfaces that are used.



Activities

To achieve your objectives, you will need to be able to do the following:

1. Configure the IP addresses for the switches (SW1=10.1.2.101/24 and SW2=10.1.2.102/24). Remember that, for switches, you assign the switch IP address to a VLAN interface (for this exercise, use VLAN 1).
2. Make sure the switches can ping each other.
3. Use the *show mac-address-table dynamic* command on each switch and verify that each switch has at least one dynamic MAC address in its **CAM** (**Content Addressable Memory**). If not, repeat the ping and try again.

4. Compare the output with the *show arp* command. You should see the same MAC address in both tables (make sure you know why you should!).
5. Issue the *show mac-address-table dynamic* command on each switch again so you can easily compare the results with the expected results.
6. Start the VPCS application by navigating to **Tools | VPCS**.
7. Assign VPC1 its IP address using the command *ip 10.1.2.11*; for example, as shown in the following command:

```
VPCS[1]> ip 10.1.2.11
Checking for duplicate address...
PC1 : 10.1.2.11 255.255.255.0
```

> While the VPC was "*Checking for duplicate address...*" it was sending gratuitous ARP requests to the MAC broadcast address, so both SW1 and SW2 should have seen a frame from VPC1 and learned the VPC1's MAC address.

8. Issue the *show mac-address-table dynamic* command on each switch again and compare the results with the preceding results. You should see a new MAC address (`0050.7966.6800`) in the MAC address table.

> By default, a MAC address should stay in the table for 300 seconds. On the NM-16ESW I was using, this seemed to be reduced to about 20 seconds, so the expected MAC addresses may have timed out. This was despite the fact that the command *show mac-address-table aging time* showed the time as 300 seconds.

9. Note the interface numbers associated with the new MAC address. On SW1, the MAC address should appear on interface **f1/0** and on SW2 on interface **f1/15**.
10. Continue assigning all IPs to the VPCS. Check the MAC address tables on your switches after each assignment.

> You need to be familiar with the `show mac-address-table` command. Be aware that on some platforms it has a slightly different syntax: `show mac address-table`. Explore the other options of this command.

11. From VPC3, start a continuous ping to VPC1 (`ping 10.1.2.11 -t`). Check both SW1 and SW2 (with the `show mac-address-table` command) to verify that the MAC address for VPC3 appears on interface **fa1/0** on SW2 and on interface **fa1/15** on SW1.
12. Shut down interface **fa1/0** (using the `shutdown` command) on SW2, where VPC3 is connected, and check the MAC address tables again. You should observe that the MAC address for VPC3 has been removed from SW2 CAM but is still present for some time on interface **fa1/15** on SW1.

> Make sure you bring interface **fa1/0** on SW2 back into service with the *no shutdown* command when you have finished.

13. Explore the concept of a broadcast domain using the following pointers:
    - Issue the *clear mac-address-table* command on SW1 and SW2.
    - Start three Wireshark live captures; one each on [SW1 **f1/0**], [SW1 **f1/1**], and [SW2 **f1/1**]. Open the Wireshark window for each of the three captures and make each fit on one-fourth of your screen, one in three corners of your screen. Place the VPCS window in the fourth corner of your screen.
    - Apply a filter of *arp* or *icmp* in each of the Wireshark captures—it is just the broadcasts and pings you want to see.
    - From VPC3, ping VPC4 (*ping 10.1.2.22*). Observe that Wireshark shows the ARP broadcast on all three screens, showing that switches pass broadcasts out all interfaces except the interface it arrived on, and that the ICMP packets are only seen on SW2 **f1/1** (VPC4).
14. Finally, observe if a destination MAC address is unknown by the switch, and the frame is flooded. You have approximately 120 seconds from the completion of the previous activity to complete this activity (before the ARP cache on VPCS times out).
    - Issue the *clear mac-address-table* command on SW1 and SW2 again. You want to see now what happens when a frame arrives at a switch when the switch doesn't know what the destination MAC address is.
    - From VPC3, ping VPC4 (*ping 10.1.2.22*). Observe that there are **no** ARP requests shown in the Wireshark captures (if there is, the ARP cache has timed out; start it again). However, you **do** see the **first** ICMP echo showing that switches pass frames with an unknown destination address from all interfaces except the interface it arrived on.

VLAN variety

VLANs are a way of restricting broadcast domains. One popular definition of a VLAN is as follows: *A VLAN is a broadcast domain.* In this exercise, you will explore how VLANs limit broadcasts, and how to route packets between VLANs.

Principle objectives

The principle objectives are designed to give you familiarity with VLAN concepts and specifically configuration on Cisco switches (from the 100-101 ICND1 Exam Topics available at `http://www.cisco.com/web/learning/exams/list/icnd1b.html#~Topics`).

- Configure and verify VLANs
- Configure and verify trunking on Cisco switches
- Describe how VLANs create logically separate networks and the need for routing between them.
- Configure SVI interfaces

Topology

The initial topology and IP addressing identical to the *Simple switching* exercise.

Activities

This set of activities will be split into the following two parts. Part 1 deals with VLAN creation and the broadcast domain concept; Part 2 extends the broadcast domain concept to multiple switches using trunk ports.

> Note that the following activities were designed to be conducted using routers with a NM-16ESW module installed. Conducting these activities on a modern Cisco switch will give different results because modern switches use Cisco's **Dynamic Trunking Protocol (DTP)** to automatically form trunk ports when switches are connected together.

Part 1

This part deals with VLAN creation and the broadcast domain concept. Complete the following steps to create broadcast domains for VLAN 10 and VLAN 20 , keeping in mind that in the beginning all ports are in the broadcast domain called VLAN 1:

1. Configure the IP addresses for the switches and the VPCs as they were in the Simple switching exercise.
2. Make sure the switches and VPCs can ping each other.
3. Create two VLANs on your switches, VLAN10 and VLAN20. Use the following commands as a guide

> The `baseconfig-sw.txt` file loads macros that turn the normal **vlan** command into a macro that creates the vlan using the **vlan database** commands. That is why the following example uses an abbreviated **vla** command—to prevent the macro from taking over. However, if this doesn't work for your version of IOS, try creating your vlans using the **vlan database** method.

```
SW1-SW#configure terminal
SW1-SW(config)#vla 10  !Note the abbreviated vlan command-
SW1-SW(config-vlan)#vlan 20
SW1-SW(config-vlan)#exit
```

4. Verify the creation of the VLANs with the *show vlan-sw* command in the privileged mode.

> On normal Cisco switches, you would use the *show vlan* command rather than the *show vlan-sw* command.

5. Now use the *switchport access vlan 10* command in the interface configuration mode on interface **f1/0** on each switch. Effectively, you have put VPC1 and VPC3 on VLAN 10.

```
SWx-SW(config)#interface f1/0
SWx-SW(config-if)#switchport access vlan 10
```

6. Observe that this action has had no effect on VPC2 and VPC4. Both VPC2 and VPC4 can still ping each other, and each can ping the switch IP addresses. But also note that now that VPC1 and VPC3 are on VLAN 10, neither VPC2 nor VPC4 can ping VPC1 or VPC3.
7. Use Wireshark captures to verify that VPC1 and VPC3 do *not* see ARP broadcasts generated by VPC2 or VPC4.
8. Also observe that VPC1 and VPC3 can't actually ping each other even though they are both on VLAN 10. You need to understand why this is so, and if the reason is not clear, it might become apparent to you in the next step.

9. Use the *switchport access vlan 20* command under the interface configuration mode on interface **f1/1** on each switch, which will put VPC2 and VPC4 on VLAN 20.
10. Another useful command to check your VLAN port configuration is *show interface status*. Make it a practice to use it often. Use it now (from privileged mode) to check if interfaces **f1/0** and **f1/1** are on the correct VLANs.
11. Observe that now that VPC1 and VPC3 are on VLAN 20, they can no longer ping each other and nor can either ping either of the switches; this is because you assigned the IP address for the switch to VLAN 1.
12. Note that the switches can still ping each other.
13. Start a Wireshark capture on the link between the two switches and open the Wireshark window for that capture. Observe that when you try and ping VPC3 from VPC1, or VPC4 from VPC2, you do *not* see the ARP broadcast appear on the link between the two switches. This is because the two inter-switch link ports are still assigned to the default VLAN 1.

Checkpoint: It is important that you realize that the reason the VPCs in the same VLAN can't ping each other is because the link between the switches is still associated with VLAN 1.

Part 2

This part extends the broadcast domain concept to multiple switches using trunk ports. At this stage, you should realize that you have three broadcast domains, VLAN 1, VLAN 10, and VLAN 20, but only VLAN 1 has connectivity between the switches. Complete the following steps to give all broadcast domains connectivity between the switches:

1. To allow traffic from multiple VLANs to traverse the inter-switch link, interface **f1/15** on each switch will have to be configured as what Cisco calls a **Trunk** port.

> On more modern Cisco switches, trunk ports are often automatically configured. Make sure you know how the `switchport mode dynamic desirable` and `switchport mode dynamic auto` commands affect the formation of trunk ports before you sit for the exam, and the role of the **Dynamic Trunking Protocol** (**DTP**) in this process (an excellent summary can be found on Brad Hedlund's blog at `http://bradhedlund.com/2007/11/27/switchport-configurations-explained/`).

2. Configure trunk ports on the **f1/15** interfaces of each switch using the following commands as a guide. Note that many switches do not require the `switchport trunk encapsulation dot1q` command because they only support one kind of VLAN trunk encapsulation.

   ```
   SW1-SW#configure terminal
   SW1-SW(config)#interface f1/15
   SW1-SW(config-if)#switchport trunk encapsulation dot1q
   SW1-SW(config-if)#switchport mode trunk
   ```

3. Observe that (after 45 seconds, when the spanning tree has stabilized) VPC1 can now ping VPC3, VPC2 can ping VPC4, and SW1 can ping SW2.
4. Next, restrict which VLANs can communicate using the *switchport trunk allowed* command. First, restrict the trunk to just VLAN 10 as in the following commands (do this on both switches):

   ```
   SW1-SW(config)#interface f1/15
   SW1-SW(config-if)#switchport trunk allowed vlan 1,10,1002-1005
   ```

5. Observe that the VPCs on VLAN 20 (VPC2 and VPC4) can no longer ping each other because VLAN 20 was not included in the list of allowed VLANs on the trunk.

> On modern switches, you will reduce the command to *switchport trunk allowed vlan 10*. On these NM-16ESW switches, you are forced to include all default VLANs in the list.

6. Use the *show interface f1/15 switchport* command to see which VLANs are allowed on the trunk.

> On modern switches, you can also use the *show interface trunk* command to check this.

7. Now add VLAN 20 to the trunks as in the following command. Note particularly the use of the term `add`.

   ```
   SW1-SW(config-if)#switchport trunk allowed vlan add 20
   ```

8. Observe that VPC2 and VPC4 on VLAN 20 can now ping each other (once the spanning tree has converged) and also verify that VPC1 and VPC3 can ping each other. In this environment (using NM-16ESW), you are *forced* to include the default VLANs on the trunks, but a common mistake on modern switches is to accidently issue the preceding command *without* the term `add`. This action results in removing *all* VLANs from the link except the VLAN listed in the command.
9. The situation in this lab at the moment is that all VPCs and switches are configured on the same IP subnet. Normally, each VLAN is on a different subnet, so reconfigure the IP addresses as shown in the following commands—note that VPC1 and VPC3 on VLAN 10 are on the `192.168.10.0/24` subnet and VPC2 and VPC4 on VLAN 20 are on the `192.168.20.0/24` subnet.

```
VPCS[1]> ip 192.168.10.11/24 192.168.10.1
VPCS[2]> ip 192.168.20.12/24 192.168.20.1
VPCS[3]> ip 192.168.10.21/24 192.168.10.1
VPCS[4]> ip 192.168.20.22/24 192.168.20.1
```

10. Now configure SW1 as a Layer 3 switch, and make it the default gateway for the VPCs. To do this on a switch that supports Layer 3 switching, use the *ip routing* command, shown as follows, in *global configuration mode*.

```
SW1-SW(config)#ip routing
```

11. Finally, assign IP addresses to the VLAN 10 and VLAN 20 interfaces, and don't forget the `no shutdown` command. Use the following model:

```
SW1-SW(config)#interface vlan 10
SW1-SW(config-if)#ip address 192.168.10.1 255.255.255.0
SW1-SW(config-if)#no shutdown
```

Validation

Verify that all VPCs can now ping each other. Also use the VPCS `trace` command to validate the path taken.

OSPF operation

For ICND1 you are expected to be able to configure and verify both OSPFv2 and OSPFv3 in a single area. For ICND2, this requirement extends to a multi area and you are also expected to demonstrate a degree of troubleshooting skills.

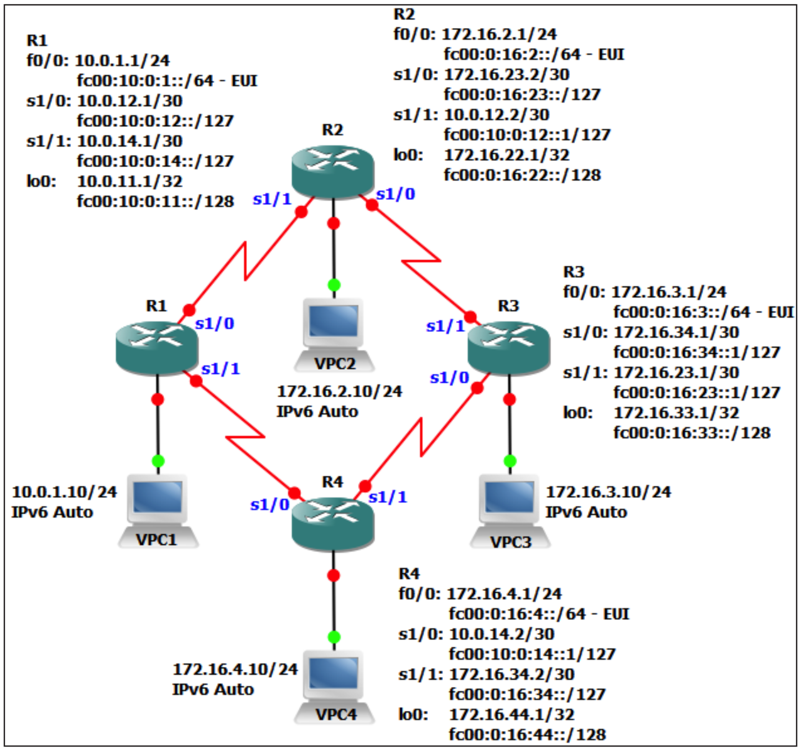
Principle objectives

The principle objectives are designed to give you familiarity with OSPF concepts and specifically configuration on Cisco routers (from the 100-101 ICND1 Exam Topics and 200-101 ICND2 Exam Topics available at `http://www.cisco.com/web/learning/exams/list/icnd1b.html#~Topics` and `http://www.cisco.com/web/learning/exams/list/icnd2b.html#~Topics`).

- Configure and verify utilizing the CLI to set the basic Router configuration
  - Cisco IOS commands to perform basic router setup
- Configure and verify OSPF (single area)
  - Configure OSPF v2
  - Configure OSPF v3

Topology

The IP addressing scheme and physical topology is shown in the following figure. It is recommended that you use C7200 routers in this topology, running version 15.x IOS.



Activities

To achieve the objectives, you will need to be able to do the following:

1. Configure IPv4/IPv6 addresses for the routers and VPCs as shown in the preceding figure.
2. Configure OSPFv2 as your routing protocol with all interfaces in area 0. Make sure your loopback interfaces are included.
3. Configure OSPFv3 as your routing IPv6 protocol with all interfaces in area 0.

Validation

Verify that all VPCs can now ping each other.

A *show ip route ospf* command on R1 should produce the following command lines:

```
R1#show ip route ospf | include /
172.16.0.0/16 is variably subnetted, 8 subnets, 3 masks
O        172.16.2.0/24 [110/65] via 10.0.12.2, 00:20:17, Serial1/0
O        172.16.3.0/24 [110/129] via 10.0.14.2, 00:20:17, Serial1/1
                       [110/129] via 10.0.12.2, 00:20:17, Serial1/0
O        172.16.4.0/24 [110/65] via 10.0.14.2, 00:20:17, Serial1/1
O        172.16.22.1/32 [110/65] via 10.0.12.2, 00:20:17, Serial1/0
O        172.16.23.0/30 [110/128] via 10.0.12.2, 00:20:17, Serial1/0
O        172.16.33.1/32 [110/129] via 10.0.14.2, 00:20:17, Serial1/1
                        [110/129] via 10.0.12.2, 00:20:17, Serial1/0
O        172.16.34.0/30 [110/128] via 10.0.14.2, 00:20:17, Serial1/1
O        172.16.44.1/32 [110/65] via 10.0.14.2, 00:20:17, Serial1/1
```

A *show ipv6 route ospf* command on R1 should produce the following command lines:

```
R1#show ipv6 route ospf | include /
O   FC00:0:16:2::/64 [110/65]
     via FE80::C801:FFF:FED8:8, Serial1/0
O   FC00:0:16:3::/64 [110/129]
     via FE80::C801:FFF:FED8:8, Serial1/0
     via FE80::C803:1FF:FE08:8, Serial1/1
O   FC00:0:16:4::/64 [110/65]
     via FE80::C803:1FF:FE08:8, Serial1/1
O   FC00:0:16:22::/128 [110/64]
```

```
     via FE80::C801:FFF:FED8:8, Serial1/0
O   FC00:0:16:23::/127 [110/128]
     via FE80::C801:FFF:FED8:8, Serial1/0
O   FC00:0:16:33::/128 [110/128]
     via FE80::C801:FFF:FED8:8, Serial1/0
     via FE80::C803:1FF:FE08:8, Serial1/1
O   FC00:0:16:34::/127 [110/128]
     via FE80::C803:1FF:FE08:8, Serial1/1
O   FC00:0:16:44::/128 [110/64]
     via FE80::C803:1FF:FE08:8, Serial1/1
```

ICND2 extensions

To achieve the ICND2 objectives, you will need to be able to do the following:

1. Modify your configurations so that all interfaces with class B IPv4 addresses are placed in area `172.16.0.0`.
2. Summarize the `172.16.0.0/16` routes being advertised by the ABRs in the most precise (longest) summary possible.
3. Summarize the `FC00:0:16::/48` routes being advertised by the ABRs in the most precise (longest) summary possible.

Validation

Verify that all VPCs can now ping each other.

A *show ip route ospf* command on R1 should produce the following command lines (note the /18 mask):

```
R1#show ip route ospf | include /
      172.16.0.0/18 is subnetted, 1 subnets
O IA     172.16.0.0 [110/65] via 10.0.14.2, 00:03:10, Serial1/1
                    [110/65] via 10.0.12.2, 00:03:32, Serial1/0
```

> Make sure you understand that by summarizing in this way, packets from R1 to, say, `172.16.44.1` will not always take the same path!

A *show ipv6 route ospf* command on R1 should produce the following command lines (note the /57 mask). On the IOS 15.1(4)M4 I was using, the `cost` parameter had to be used on the `area … range` command to get two summary routes.

```
R1#show ipv6 route ospf | include /
OI  FC00:0:16::/57 [110/193]
     via FE80::C801:FFF:FE14:8, Serial1/0
     via FE80::C802:FFF:FEB0:8, Serial1/1
```

EIGRP excitement

EIGRP is not a requirement for ICND1, but make sure you fulfill the following objectives for ICND2.

This lab starts by configuring OSPF so you can observe the effect of running two routing protocols with different administrative distances.

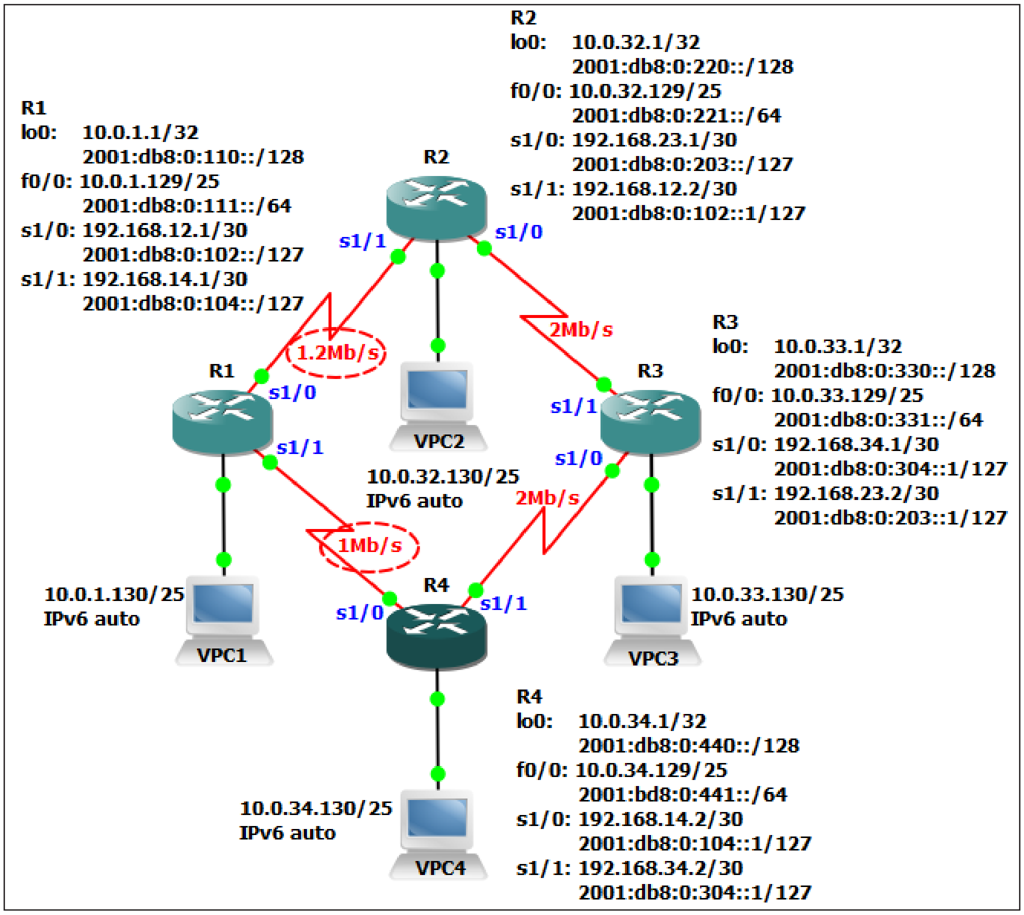
Principle objectives

The principle objectives are designed to give you familiarity with EIGRP concepts and specifically configuration on Cisco routers (from the 200-101 ICND2 Exam Topics available at `http://www.cisco.com/web/learning/exams/list/icnd2b.html#~Topics`).

- Configure and verify EIGRP (single AS)
    - Feasible Distance / Feasible Successors / Administrative distance
    - Feasibility condition
    - Metric composition
    - Router ID
    - Auto summary
    - Path selection
    - Load balancing
    - Equal
    - Unequal
    - Passive interface

Topology

The IP addressing scheme and physical topology is shown in the following figure. It is recommended that you use C7200 routers in this topology, running version 15.x IOS.



Activities

To achieve the objectives, you will need to be able to do the following:

1. Configure the IPv4/IPv6 addresses for the routers and VPCs as shown in the preceding figure.
2. Make sure you include the bandwidth statements on the serial interfaces, and note that the link between R1 & R4 is 1 Mb/s and between R1 & R2 it is 1.2 Mb/s.
3. Configure OSPFv2 as your routing protocol with all interfaces in area 0. Make sure your loopback interfaces are included. Later, you will configure EIGRP and observe that the EIGRP routes replace the OSPF routes.

4. Configure OSPFv3 as your routing IPv6 protocol with all interfaces in area 0.
5. Verify that routing has converged and that all VPCs can ping each other (IPv4 & IPv6) as well as ping all loopback addresses.
6. Capture a copy of the routing tables (IPv4 and IPv6) and save it in a text editor. Note particularly the administrative distance of the OSPF routes, and make sure you know how to identify the administrative distance in the output of the show `ip` route, show `ipv6` route, and show ip `protocols` commands.
7. Verify (using the VPCS `trace` command) that the path taken between VPC1 and VPC3 is via R2 (because the bandwidth is greater).
8. Begin a continuous (IPv4) ping between VPC1 and VPC3 and then shut down one of the serial interfaces on R2 to force OSPF to reconverge. Note how long the convergence takes. Repeat for an IPv6 ping.
9. Don't forget to bring the interface back into service (using the no `shutdown` command).
10. Configure EIGRP as your routing protocol for AS 65500. Make sure your loopback interfaces are included.
11. Verify that the routing has converged using the *show ip route* and *show ipv6 route* commands.
12. Compare the routing tables now with the output you saved in a text editor (back in step 6). You should observe that the routing tables are exactly the same, but now the EIGRP routes have replaced the OSPF routes. Note the administrative distance of the new EIGRP routes is lesser than the old OSPF routes.
13. If all of the OSPF routes have disappeared, remove OSPF routing. If not, troubleshoot (you did remember to use the *no auto-summary* command, didn't you?).
14. Verify (using the VPCS `trace` command) that the path taken between VPC1 and VPC3 is via R2 (because the bandwidth is greater).
15. Verify (using the *show ip eigrp topology* and *show ipv6 eigrp topology* commands) that R1 holds a **feasible successor** route to R3. Make sure you understand how to extract this information from the output.
16. Begin a continuous (IPv4) ping between VPC1 and VPC3 and then shut down one of the serial interfaces on R2 to force the EIGRP to reconverge. Note how long the convergence takes. Repeat for an IPv6 ping.
17. Don't forget to bring the interface back into service (using the *no shutdown* command).

18. Repeat the failover test with debugging turned on (using the *debug eigrp fsm* command).
19. Change the **variance** so that traffic between R1 and R3 is distributed between R2 and R4.
20. Observe the effect that this has on your routing table.
21. Begin a Wireshark capture on interface **f0/0** on R1. Observe that the EIGRP hello packets are sent every five seconds for both IPv4 and IPv6.
22. Now modify your EIGRP (IPv4) configuration to make interface **f0/0** on **R1** a **passive interface** and observe the result in Wireshark. Repeat for IPv6.
23. Add the *auto-summary* command to your IPv4 EIGRP configuration. Note the effect this has on your configuration and make sure you understand why VPC1 can no longer (IPv4) ping VPC3 and why the routing table does not show any `/24` masks for the `10.0.0.0` network.

> Before IOS 15.x, `auto-summary` was enabled by default.

24. Summarize the `10.0.0.0/8` routes being advertised to R1 by R2 and R4 in the most efficient (longest) single summary route possible.
25. Summarize the `2001:DB8::/48` routes being advertised to R1 by R2 and R4 in the most efficient (longest) single summary route possible.

Validation

After the completion of this exercise, R1's EIGRP routing tables should look like the following command lines:

```
R1#show ip route eigrp | include /
    10.0.0.0/8 is variably subnetted, 10 subnets, 3 masks
D  10.0.32.0/22 [90/2647808] via 192.168.12.2, 00:31:35, Serial1/0
      192.168.23.0/30 is subnetted, 1 subnets
D  192.168.23.0 [90/3157248] via 192.168.12.2, 04:25:55, Serial1/0
      192.168.34.0/30 is subnetted, 1 subnets
D  192.168.34.0 [90/3584000] via 192.168.14.2, 04:25:53, Serial1/1
R1#show ipv6 route eigrp | include /
D   2001:DB8::/53 [90/2647808]
     via FE80::C83D:19FF:FEB4:8, Serial1/1
     via FE80::C83C:1DFF:FEEC:8, Serial1/0
```

Extending to CCNP certification

CCNP certification requires a much deeper understanding of the actual protocols than CCNA does. These labs are therefore not so prescriptive but more oriented toward guiding you to create your own labs. You will find that using GNS3 to implement your own designs will give you a far more real-life experience than trying to work with someone else's ideas.

EIGRP

A solid understanding of EIGRP is essential for CCNP.

Principle objectives

The principle objectives are designed to give you a thorough understanding of EIGRP concepts and specifically configuration on Cisco routers (from the 200-101 642-902 ROUTE Exam Topics available at `http://www.cisco.com/web/learning/exams/list/route.html#~Topics`).

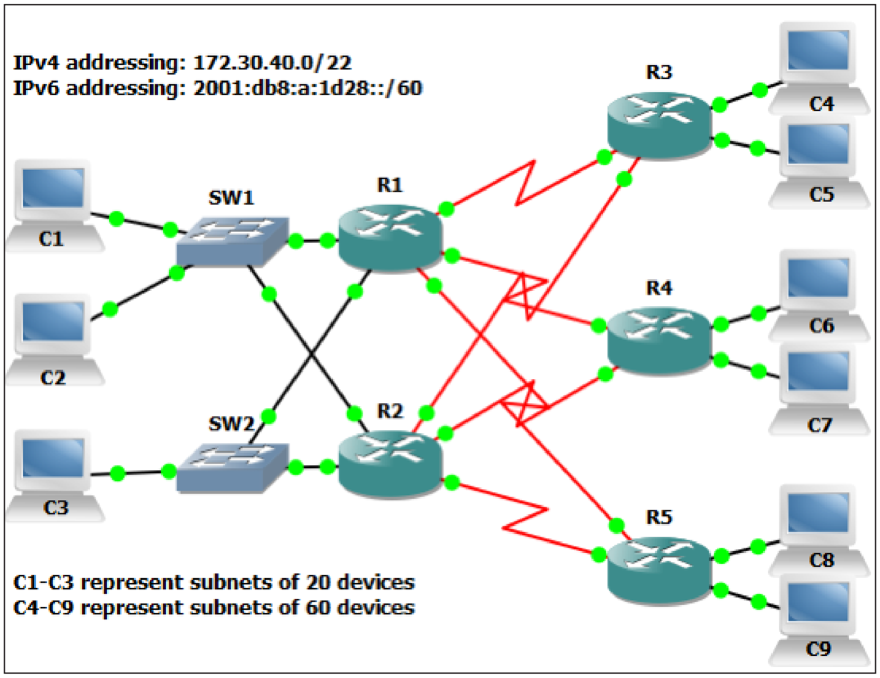
- Determine network resources needed for implementing EIGRP in a network
- Create an EIGRP implementation plan
- Create an EIGRP verification plan
- Configure EIGRP routing
- Verify if an EIGRP solution was implemented properly using the `show` and `debug` commands
- Document the verification results for an EIGRP implementation
- Create an IPv6 implementation plan
- Create an IPv6 verification plan
- Configure IPv6 routing

Topology

Your network consists of a central office with two core routers, R1 and R2. Three new branch offices are being added, with dual connections back to the central office—one link each to R1 and R2. The branch offices each have two subnets that are required to support 60 users each. A third subnet for BYOD wireless access is being planned for each branch office that will need to support up to 100 devices. You must make provision for this subnet in your plan.

In accordance with the rest of the company's addressing scheme, you have been allocated the IP address spaces of `172.30.40.0/22` and `2001:db8:a:1d28::/60` to number your network, and you are expected to implement the addressing scheme in a structured way so that summary routes can be advertised from the branch offices to the core.

At the network core, two switches, SW1 and SW2, support three more subnets of up to 20 devices each and have trunk ports connected to R1 and R2. These subnets are all configured for FHRP on R1 and R2. The physical layout can be represented as shown in the following figure:


IPv4 addressing: 172.30.40.0/22
IPv6 addressing: 2001:db8:a:1d28::/60
R3
C4
C5
C1
SW1
R1
R4
C6
C2
C7
SW2
R2
C3
R5
C8
C1-C3 represent subnets of 20 devices
C4-C9 represent subnets of 60 devices
C9

Activities

To achieve the objectives, you will need to be able to do the following:

1. Create an implementation plan for this network, including practical suggestions for the router and switch models. Make sure you include additional network modules that might be required for these routers.
2. Part of your plan must include a verification plan—a list of the tests you intend to carry out to prove that the implementation has been completed. Make sure it includes validation for the following:
    - Full connectivity (IPv4 and IPV6)
    - The FHRP operation
    - DHCP address allocation
    - Route summarization
3. Create and document an IP addressing plan (IPv4 and IPv6) based on the allocated address spaces of `172.30.40.0/22` and `2001:db8:a:1d28::/60`. Make sure your plan follows a logical scheme that can be expanded in the future.
4. Build a GNS3 simulation that reflects your network. You may not be able to use exactly the same router models that you proposed in Activity 1, but you can still implement the addressing plan. I suggest using C7200 routers running IOS15.x.
5. Remember that core network subnets must support a FHRP. Ensure this is implemented for IPv4 hosts by configuring DHCP on the core routers.
6. Ensure EIGRP hellos are not seen on PC subnets.
7. Summarize routes wherever possible, demonstrating your summarization-addressing scheme.
8. Carefully document your topology, including all IP addressing and all summary routes used.

9. If you your network, take the opportunity to practice your troubleshooting skills by documenting the steps you took to do the following:
    - Identify the problem
    - The possibilities you considered
    - The steps you took to test these possibilities
    - The steps you took to validate that you had implanted the correct solution

Validation

Complete the verification test plan you created in Activity 1.

Multi-area OSPF

A solid understanding of OSPF is essential for CCNP.

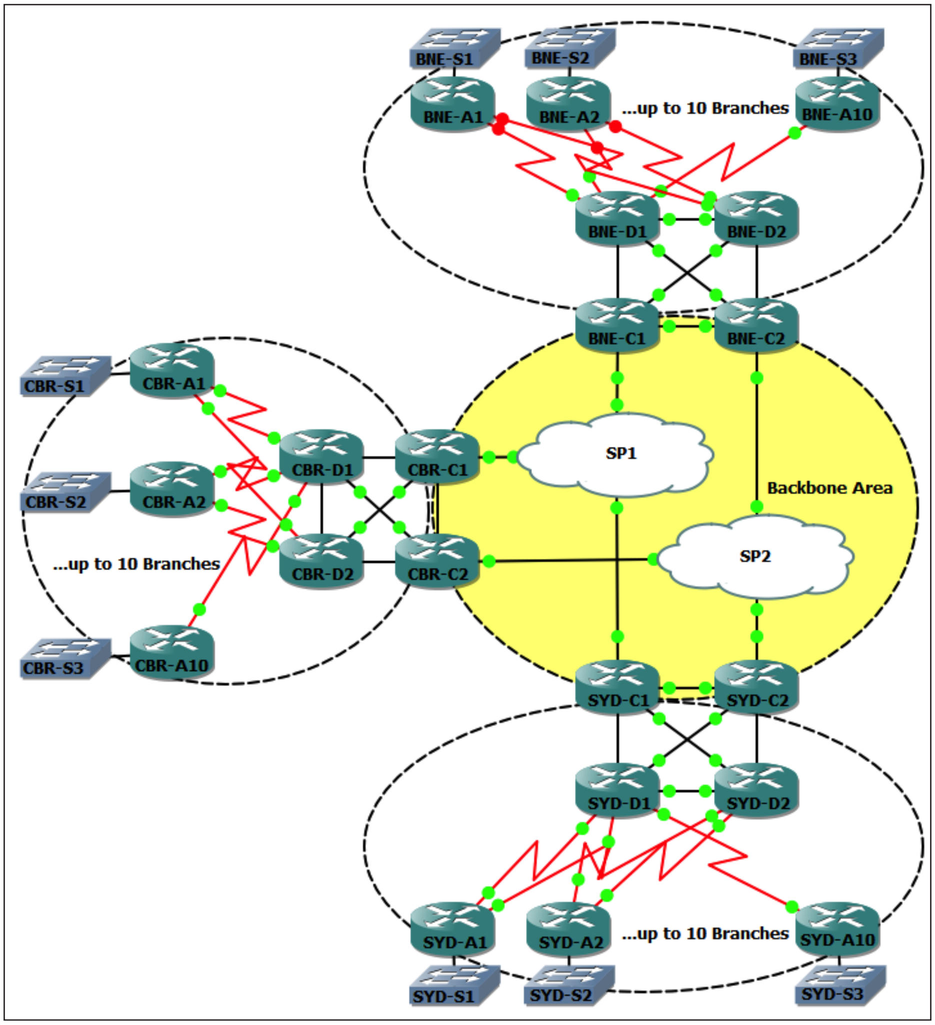
Principle objectives

The principle objectives are designed to give you a thorough understanding of OSPF concepts and specifically configuration on Cisco routers (from the 200-101 642-902 ROUTE Exam Topics available at `http://www.cisco.com/web/learning/exams/list/route.html#~Topics`).

- Determine the network resources needed for implementing OSPF on a network
- Create an OSPF implementation plan
- Create an OSPF verification plan
- Configure OSPF routing
- Verify if the OSPF solution was implemented properly using the `show` and `debug` commands
- Document the verification results for an OSPF implementation plan

Topology

Your company has a national network with dual core routers in three regional cities across the country. Each city has a data center where two distribution routers provide access to up to 10 branch offices, which may have single or dual links to the regional office. Each branch has a LAN attached, representing up to 250 users.



The clouds marked **SP1** and **SP2** are generic Ethernet switch devices but with the symbol changed (by navigating to **Device** | **Change Symbol**).

The physical layout can be represented as in the preceding figure. Each office, **SYD**, **BNE**, and **CBR,** has 100 Mb/s Ethernet links to two service providers, **SP1** and **SP2**. Each office also has a cluster of two core routers and two distribution routers fully meshed with Ethernet connections. Finally, the distribution routers have connectivity to the branch routers via serial links.

> In GNS3, it is often a good idea to use serial links even if you use Ethernet links in the same situation in the real world. This is because serial links behave more like directly connected routers—if you shut down one end of the serial link, the other end goes down too. Due to a shortcoming of Dynamips, shutting down one end of an Ethernet link does not bring down the other.

Activities

To achieve the objectives, you will need to be able to do the following:

1. Create an implementation plan for this network, including practical suggestions for the router and switch models. Make sure you include additional network modules that might be required for these routers.
2. Part of your plan must include a verification plan—a list of the tests you intend to carry out to prove that the implementation has been completed. Make sure it includes validation for:
    - Full connectivity (IPv4 and IPV6)
    - High availability
    - Route summarization
3. Design and document an IPv4 and IPv6 addressing scheme for this network based on RFCs 1918 and 4193. Organize your addressing so that routes from each OSPF area can be summarized on the ABRs. Make sure your plan includes addresses for loopback interfaces, and follows a logical scheme that can be expanded in the future.
4. Build a GNS3 simulation that reflects your network. You may not be able to use exactly the same router models that you proposed in Activity 1, but you could still implement the addressing plan. I suggest using C7200 routers running IOS 15.x.

> You don't have to build the whole network as shown to build a simulation that reflects this network, especially if your hardware is a little light on. You could simulate this network using one SP switch, and for each site have one core, one distribution, and one access router. For simulating multiple routes, you can make one access router simulate ten branches by creating ten loopback interfaces and assigning them the IP addresses you would have assigned the branch office LANs.

5. Ensure OSPF hellos are not seen on PC subnets.
6. Summarize routes where possible, demonstrating your summarization-addressing scheme.
7. Carefully document your topology, including all IP addressing (including loopback addresses) and all summary routes used.
8. If you face issues while building your network, take the opportunity to practice your troubleshooting skills by documenting the following steps you took to:
    - Identify the problem
    - The possibilities you considered
    - The steps you took to test these possibilities
    - The steps you took to validate that you had implanted the correct solution

Extensions

You might like to try the following extensions:

- Change the routing protocol between the CBR branch offices and the **CBR-D1** and **CBR-D2** routers to RIPv2. Redistribute the RIP routes into OSPF at the **CBR-D1** and **CBR-D2** routers and redistribute a default route into RIPv2.
- Change the routing protocol between the BNE branch offices and the **BNE-D1** and **BNE-D2** routers to EIGRP. Redistribute the EIGRP routes into OSPF at the **BNE-D1** and **BNE-D2** routers and redistribute OSPF summary routes into EIGRP.
- Change the core-network-SP-facing interfaces to IPv6 only and tunnel the IPv4 traffic between the regional offices using GRE tunnels.

Validation

Complete the verification test plan you created in Activity 1.

eBGP

For CCNP purposes, your understanding of BGP is limited to eBGP, which is best understood in the context of an enterprise wishing to dual-home its Internet connections to two different service providers on a single router. Normally, you would use two routers, but that would require the use of iBGP, which is not on the CCNP ROUTE exam objectives.

Principle objectives

The principle objectives are designed to give you a thorough understanding of BGP concepts and specifically configuration on Cisco routers (from the 200-101 642-902 ROUTE Exam Topics available at `http://www.cisco.com/web/learning/exams/list/route.html#~Topics`).

- Determine network resources needed for implementing eBGP on a network
- Create an eBGP implementation plan
- Create an eBGP verification plan
- Configure eBGP routing
- Verify if the eBGP solution was implemented properly using the `show` and `debug` commands
- Document verification results for an eBGP implementation plan

Topology

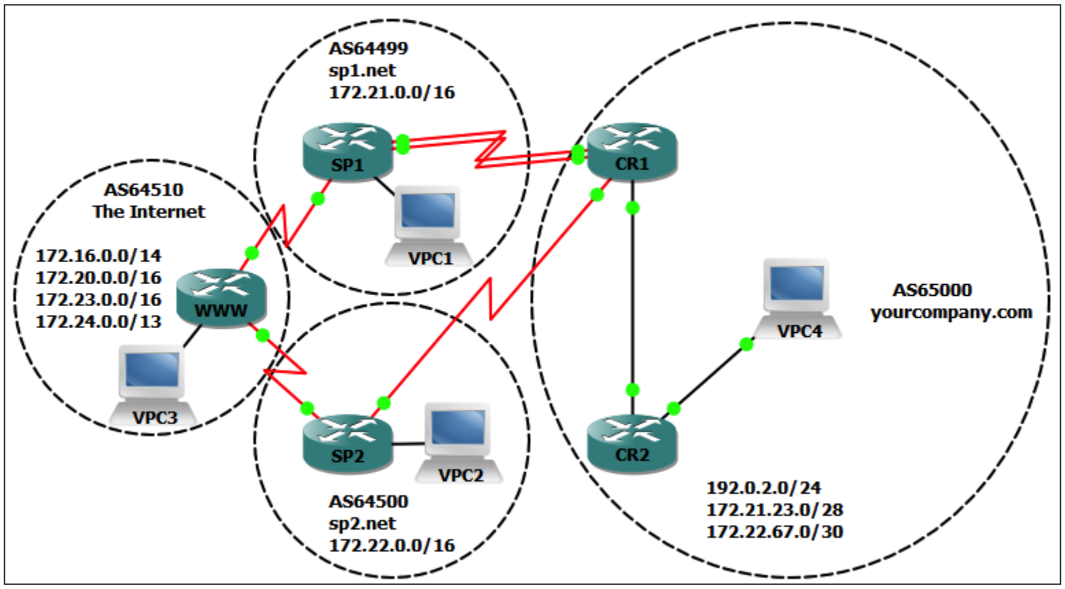
Your company (**yourcomapany.com**) has dual connections to two service providers via a router that is currently using static default routes to provide connectivity. Your job is to migrate your organization to exchange eBGP updates for high-availability purposes.

In the following diagram, **AS64510** represents the Internet and the router **WWW** originates a few prefixes that it advertises to **SP1** and **SP2**. These prefixes represent the thousands of prefixes you would receive from the Internet if you did peer with your service provider.

AS64499 is a service provider and originates routes for the `172.21.0.0/16` prefix on router **SP1** while **AS64500** originates routes for the `172.22.0.0/16` prefix on the router **SP2**.

These service providers have allocated **yourcompany.com** prefixes of `172.21.23.0/28` and `172.22.67.0/30` to use for peering with **SP1** and **SP2** respectively. The rest of **yourcompany.com** uses the `10.0.0.0/8` address space internally.

This exercise will give you practice in configuring eBGP not only between **yourcompany.com** to **SP1** and **SP2**, but also between the service provider routers and the **WWW** router.



Activities

To achieve your objectives, you will need to be able to do the following:

1. Create an implementation plan for this network, including practical suggestions for router and switch models. Make sure you include additional network modules that might be required for these routers.
2. Your plan must address the following:
   - eBGP multi-hop between **CR1** and **SP1**
   - All traffic to the **WWW** advertised routes and the `172.21.0.0/16` network is to be via **SP1**
   - Normally, only traffic destined for `172.22.0.0/16` should exit **yourcompany.com** via **SP2**; except…
   - If both links between **CR1** and **SP1** are down, all traffic should be routed via **SP2**

    - Similarly, if the link between **CR1** and **SP2** is down, traffic to `172.22.0.0/16` should exit **yourcompany.com** via **SP1**
    - Under **NO** circumstances should traffic between the **SP1**, **SP2**, and **www** routers ever be routed via **CR1**

3. A part of your plan must include a verification plan—a list of the tests you intend to carry out to prove that the implementation has been completed. Make sure it includes validation for the following:
    - Correct routing (`172.22.0.0/16` via **SP2** and all other traffic via **SP1**)
    - Failover for a single link failure between **CR1** and **SP1**
    - Failover for complete failure between **CR1** and **SP1**
    - Failover for complete failure between **CR1** and **SP2**
    - Access to **WWW** routes are maintained even if the link between **SP1** and **WWW** goes down
    - No pass-through service provider or Internet traffic.
4. Design and document an IPv4 addressing scheme for this network based on the addressing given.
5. Build a GNS3 simulation of the network. You may not be able to use exactly the same router models that you proposed in your implementation plan, but you could still implement the addressing plan. I suggest using C7200 routers running IOS15.x.

Validation

Complete the verification test plan you created in the preceding activities.

Striving for the "Holy Grail" – how to use GNS3 to prepare for the CCIE R&S lab

There is no doubt that GNS3 can be a brilliant tool to help tune your skills for CCIE—especially the **Routing and Switching** (**R&S**) version. But be aware that you will not be able to do **everything** you need to practice, especially when it comes to switching. What I would suggest you do to make the most of GNS3 for your CCIE R&S studies, is to build a GNS3 lab using simulated switches to practice exercises that are routing oriented, and also build another lab with real switches—preferably Catalyst 3560s.

Cisco doesn't tell us exactly what the physical layout of the lab will be, so you have to come up with your own design that will allow you to test all of the features you are expected to know, as outlined in the CCIE R&S blueprint. The CCIE R&S blueprint can be found at `https://learningnetwork.cisco.com/docs/DOC-4375` (a Cisco login is required.)

The following lab designs are suggestions only, and you may end up with several IP addressing schemes to cater for different types of labs. The beauty of GNS3 is that you can quickly build a new lab for any scenario you wish—you don't have to stick to a rigid setup such as those found in hardware-based labs. On the other hand, you should still try and make your lab similar to what you will expect. For instance, I would expect that all routers would be connected to switch ports—no direct Ethernet links between routers. Once you have all your routers connected to a core-switched network, you can use the switches as "electronic patch panels" using VLANs to connect any router to any other router.

Start with some switches—layer 3 switches so you can practice SVI configurations. You'll require two switches at least to do a decent job—four if you are using hardware switches and want to practice **Spanning Tree**, **Link Aggregation**, or other Layer 2 features.

You will need a frame-relay switch, and some examples of direct serial connections that you can use for both frame relay and PPP connections. Have two routers connected with two parallel serial links so you can build MPPP and eBGP multi hop topologies.

Next, add enough routers to give you a decent sized network to allow any variation you can think of.

You probably won't find too many PCs in the CCIE lab exam, so the use of VPCs is optional. For me, I like to have something beyond a router interface that I can ping on a stub network, so I have added some virtual PCs to the mix. Again, because they are connected to switches, I can easily have them on any VLAN connected to any router I care to.

Many people striving for CCIE buy commercial lab designs, but in many ways you will actually get more out of the exercises if you go through the thinking process of designing your own labs.

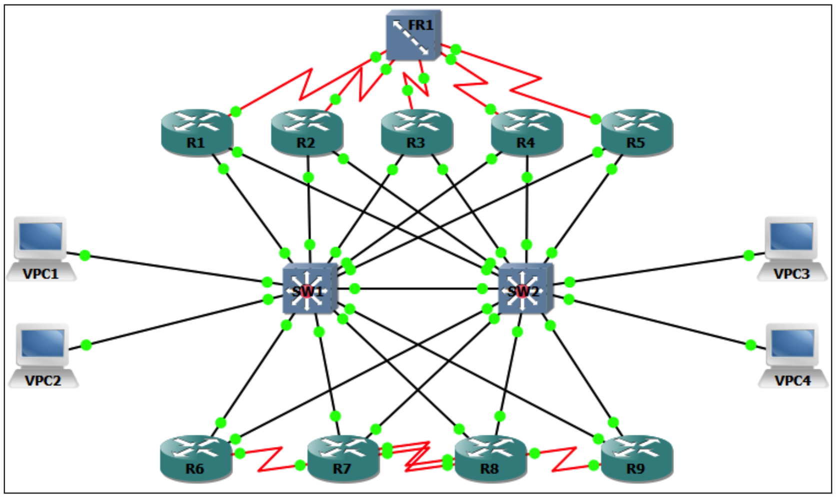
Sample CCIE practice lab design using simulated switches

This design uses two c3725 routers with NM-16ESW modules added. They are used purely as layer 3 switches. The **f0/0** and **f0/1** interfaces are not used. They are used to provide VLAN paths between all of the Ethernet interfaces on all the other routers. They also serve as fully functioning routers using SVIs, but are of limited use for exploring the more sophisticated switching labs.

Topology

The following is one possible topology. To make the layout easy to remember, there are various patterns I use:

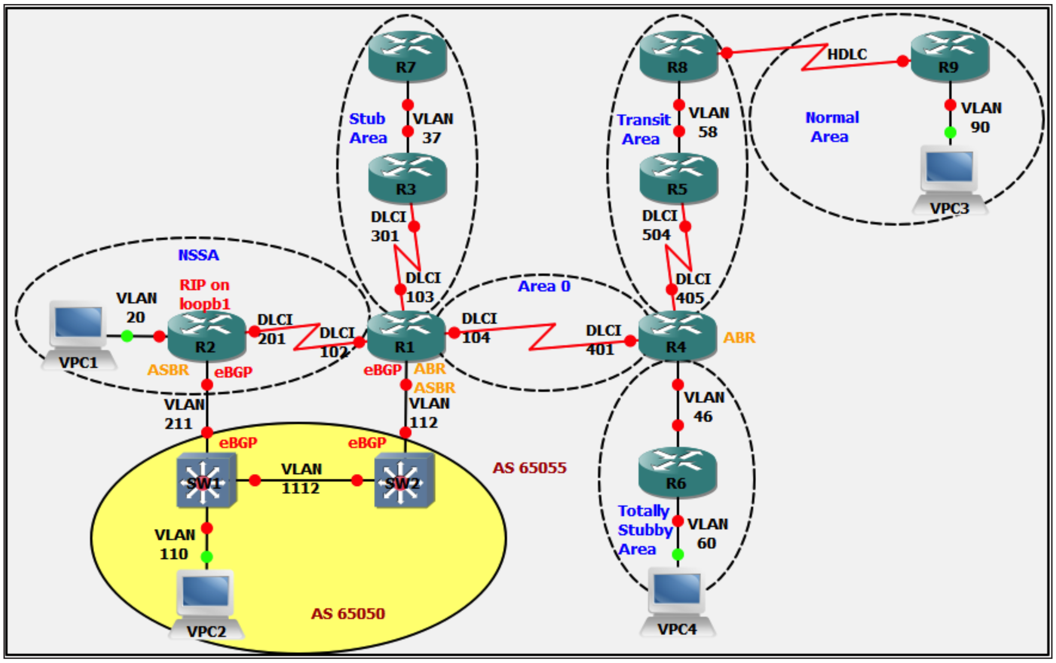
- **R1 s1/0** connects to **FR1** port **1**; similarly, **R2** to **R5** connects to **FR1** ports **2** to **5**.
- **R1 s1/0** has access to four DLCIs, numbered `102` to `105` that switch **FR1** switches to port/DLCIs **2/201**, **3/301**, **4/401**, and **5/501** respectively. Similarly for **R2** to **R5**.
- **R1 f0/0** connects to **SW1 f1/1**, and **f1/0** connects to **SW2 f1/1**.
- The pattern for Ethernet interfaces is that router **R***n* **f0/0** connects to **SW1 f1/***n*, and **f1/0** connects to **SW2 f1/***n*; for example **R***1* **f0/0** connects to **SW1 f1/***1* **f1/0** connects to **SW2 f1/***1*, and so on through to **R9**.
- VPC*n* connects to a port number of **f1/10+***n*, so VPC2 connects to **SW1 f1/12** and VPC3 connects to **SW2 f1/13**.



Activities

Using the preceding topology, (remember the switches can be used as routers), design a network that includes the following features:

1. Six OSPF areas, including the backbone area, one normal area, one stub area, one totally stubby area, one NSSA, and a transit area.
2. BGP connections from the same AS to two ASBRs, one in area 0, and one in the NSSA.
3. The ASBR connecting to the NSSA is to also redistribute a set of RIPv2 routes. Configure multiple networks on, say, the interface **loopback1** to generate the RIP routes.
4. You will not need to use all of the interfaces that are available. Shut down the interfaces that you don't need to use.
5. The idea is that you use the layout you have to build any kind of network you want. The logical view of your network might end up looking like the following figure:



Note that although you *could* have just built the topology shown in the preceding figure from scratch, you *will* almost certainly have to use VLANs and DLCIs in the CCIE R&S lab to connect different parts of the lab to other parts of the lab. As a CCIE candidate you will already be very familiar with the idea that a VLAN or a frame-relay DLCI pair are just pieces of cable that you can use to join any two connected devices.

Note that the VLAN numbers used have logic as well. Any VLAN connecting two routers has a VLAN ID made up of the two router numbers (lower number first). The switches assume router numbers 11 and 12. Stub VLANs (VLANs that don't connect to another router) take the router number followed by a zero (such as VLAN 20). If a second stub VLAN were required, it would be the router number twice (such as VLAN 22). The plan is not foolproof when there are more than 9 routers or when more than two routers share a subnet, but it serves as a guide.

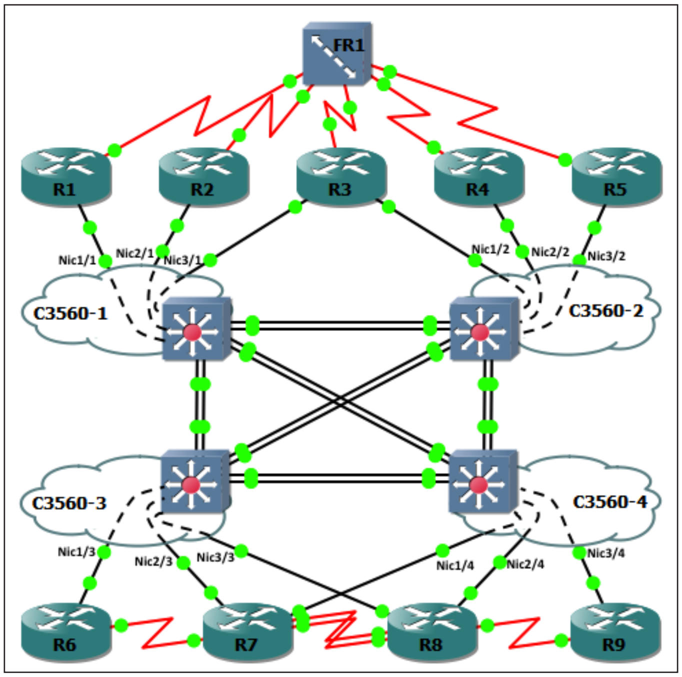
Of course, to complete the lab you will need an IP addressing scheme. Again, to make this easy to remember, try and use the same logic to assign subnet numbers if possible. If you are attempting some logical route summarization, this may not work, but if you use the `10.0.0.0/8` address space, you can use a scheme where octet #2 represents an area number and octet #3 represents the router numbers. The lowest numbered router always gets the lowest IP of the subnet, and then IP addresses increment from there in router-number order. For instance, using this scheme, the link between R1 and R4 above would give **R1** an IP of `10.0.14.1/30` and **R4** an IP of `10.0.14.2/30`.

Sample CCIE practice lab design using your own switches

Many people believe that the best use of GNS3 when preparing for CCIE R&S is to use a set of four hardware switches (Catalyst 3560s if possible—3550s will do most of what you need) and then run GNS3 on a host computer with multiple NICs. You will need one NIC for each physical connection you want from your GNS3 topology—so if you want to have, say three, devices connected to each of four switches, you will need twelve NICs. You are sure to want to run 802.1Q VLANs on these ports, so make sure you choose NICs that support 802.1Q and make sure your host computer is running an operating system that supports 802.1Q VLAN tags. Quad port NICs are popular for this purpose, and Linux is a popular choice of OS. The GNS3 website has some more information about recommended NICs and OS at `http://www.gns3.net/switching-quad-nic-pci-network-cards-option/`.

You can now run multiple cables from your multiple NICs to multiple ports on your set of switches.

In GNS3, you add these interfaces as cloud devices. You can allocate multiple interfaces to one cloud to reduce the number of cloud icons on your screen. I would suggest that you configure one cloud for each physical switch you have, and then allocate a number of ports to each switch. If you had three quad NICs, you would get twelve ports, so allocate three ports to each switch—possibly port 1 on each NIC to one switch, port 2 on each NIC to another switch, and so on. The end result should look something like the following figure, with the dotted lines representing the cables you would use to connect your host computer's multiple NICs to the physical switches, which are represented by the four switch icons in the middle of the figure:



Note that you now have a reduced number of interfaces you can connect your routers to, even with three quad NICs. But remember, if you configure physical switch ports as 802.1Q trunk ports, you can run multiple VLANs that can then connect to multiple sub interfaces on your routers.

Remember, the four switch icons in the middle diagram above are not GNS3 switches, but a representation of how you would connect your physical switches so that you can practice spanning-tree, link aggregation (port-channels), and other Layer 2 features.

Summary

Whether you are starting out with CCENT-level certification or aiming for CCIE, you will now have more ideas for not only how you can use GNS3 to practice your skills, but also some useful tips on designing a lab environment. You should now be equipped to design your own exercises and maximize your study efforts. Even if you are not studying for certifications, you should now be able to set up a lab environment for any simulation.