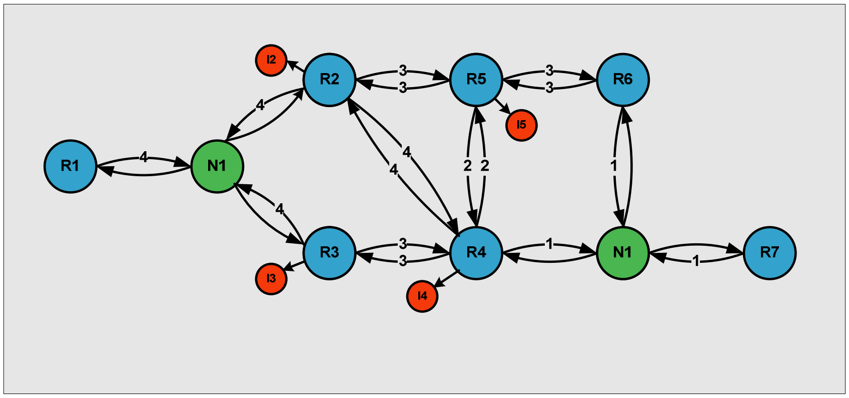
NETMASTERCLASS
ROUTING AND SWITCHING CCIE® TRACK

DOiT-200v6
VOLUME II



Scenario 7
ANSWER KEY

FOR

CCIE® CANDIDATES

☎ 1-888-677-2669 http://www.netmasterclass.net 13530 Dulles Technology Drive, Suite #150, Herndon, VA 20171

Disclaimer

NetMasterClass, LLC is an independent training and consulting company based in Herndon, Virginia. The terms “Cisco”, “Cisco Systems” and “CCIE” are the trademarks of Cisco Systems, Inc. NetMasterClass, LLC is Cisco Learning Partner.

Cisco Non-Disclosure Agreement Compliance

All products and services offered by NetMasterClass, LLC are in full compliance with the Cisco CCIE Lab non-disclosure agreement. The content of the NetMasterClass CCIE preparation materials is based upon the NetMasterClass “issue spotting and analysis” internetwork training methods.

NOTE: To use this document to its maximum effectiveness, access the SHOWiT engine while reviewing each section of this Answer Key.

DOiT-V6 Scenario 7: Spot the Issue Answer Key

Table of Contents

Attention!

REGARDLESS OF ANY CONFIGURATION YOU PERFORM IN THIS EXAM, IT IS VERY IMPORTANT TO CONFORM TO THE GENERAL GUIDELINES PROVIDED BELOW. IF YOU DO NOT CONFORM TO THEM, THIS CAN RESULT IN A SIGNIFICANT DEDUCTION OF POINTS IN YOUR FINAL EXAM SCORE.

Goals and Restrictions

- IP subnets displayed in the Scenario diagram belong to network 172.16.0.0/16.
- Do not use any static routes.
- Do not rely on dynamic Frame-Relay Inverse ARP.
- Make sure all IPv4 and IPv6 loopback interfaces are advertised with their original mask, unless noted otherwise.
- Make sure all IPv4 interfaces in the diagram are ***reachable*** within this internetwork. **DO NOT FORGET THIS!**
- Do not use “ip default-network”.
- All IP address involved in this scenario must be reachable, unless specified otherwise.
- Networks advertised in the BGP section must be reachable only in the BGP domain.

Explanation of Each of the Goals and Restrictions:

IP subnets in the Scenario diagram belong to network 172.16.0.0/16

The third and forth octets of the IP addresses displayed on the diagram belong to 172.16.0.0/16.

Do not use any static routes.

Static routes can be used to solve a range of reachability problems. However, you cannot use them. You must rely on skillful configuration of all your unicast routing protocols.

Do not rely on dynamic Frame-Relay Inverse ARP.

This requirement forces you to fulfill your Frame-Relay inverse arp requirements with Frame-Relay map statements. Think of a Frame-Relay map statement as the equivalent of a static inverse arp entry.

Make sure all IPv4 and IPv6 loopback interfaces are advertised with their original mask, unless noted otherwise.

This requirement is primarily for the OSPF advertised loopbacks. Use “ip ospf network point-to-point” under the loopback interface. Otherwise, the loopback will be advertised as a /32 host entry by default.

Make sure all IP interfaces in the diagram are *reachable* within this internetwork. DO NOT FORGET THIS!

This is a key goal to observe. This requires that all of your IGP’s are configured properly. Also, all of your routing policy tasks must be configured properly. The key elements of your routing policy include route redistribution and the controlling of routing updates using distribute-lists, route-maps and the distance command. A key point to remember about this exam is: the term “redistribution” is never explicitly used in this exam. However, you must perform redistribution in order to assure that all IP addresses are reachable without the use of static routes.

Use conventional routing algorithms.

This restriction prevents you from solving any problems by configuring policy routing. At the heart of this restriction is the interpretation of the “conventional routing algorithms”. Although this phrase can be interpreted in a number of different ways, the interpretation applied in this workbook is:

> CONVENTIONAL ROUTING ALGORITHMS ARE ROUTING ALGORITHMS THAT APPLY DESTINATION BASED PREFIX LOOKUPS IN A ROUTING TABLE. CONVENTIONAL ROUTING ALGORITHMS DO NOT USE ANY OTHER TYPE OF INFORMAITON OTHER THAN THE DESTINATION ADDRESS TO MAKE A PACKET FORWARDING DECISION.

Due to this restrictive interpretation, no form of policy routing can be applied. Whenever you see this restriction, you will need to use dynamic routing protocols to fulfill all packet forwarding requirements

The following IOS versions were used on the devices:

Device	IOS version
R1	IOS (tm) 3600 Software (C3640-JK9O3S-M), Version 12.4(3)
R2	IOS (tm) 3600 Software (C3640-JK9O3S-M), Version 12.4(3)
R3	IOS (tm) C2600 Software (C2600-J1S3-M), Version 12.3(15a)
R4	IOS (tm) C2600 Software (C2600-J1S3-M), Version 12.3(15a)
R5	IOS (tm) 3600 Software (C3640-JK9O3S-M), Version 12.4(3)
R6	IOS (tm) 3600 Software (C3640-JK9O3S-M), Version 12.4(3)
FRS	IOS (tm) 2500 Software (C2500-JS-L), Version 12.2(27)
CAT1	IOS (tm) C3550 Software (C3550-I5Q3L2-M), Version 12.2(25)SEA
CAT2	IOS (tm) C3550 Software (C3550-I5Q3L2-M), Version 12.2(25)SEA

7.1 Frame Relay

HIDDEN ISSUES TO SPOT WITH FRAME RELAY

Issue: Configure the minimal number of PVC's for subnet 172.16.234.0/24.

Solution:

Even though the network diagram appears to display a hub and spoke topology on the 172.16.234.0/24 subnet, do not be deceived. You must read ahead to the IGP section and determine what IGP is being assigned to this subnet. After reading the IGP section, you will discover that OSPF is running over the 172.16.234.0/24 subnet. In order for OSPF to have any router to become a DR on area 234 and operate properly, all routers on the same subnet must maintain an adjacency to each other. Therefore, all routers on this subnet need a direct connection to each other. As a result, a full mesh of Frame-Relay PVC's must be configured on the 172.16.234.0/24 subnet.

Implementation:

See the solution for the next issue.

Verification:

See the verification for the next issue.

Issue: Configure logical interfaces on the 172.16.234.0/24 subnet.

Solution:

Since a full mesh is required on the 172.16.234.0/24 subnet, two DLCI's will be used on each of the routers attached to this subnet. Since two DLCI's are used on each router for this subnet a point-to-point subinterface cannot be used. Since you must use a logical interface, you must use a multipoint subinterface on all Frame-Relay interfaces attached to the 172.16.234.0/24 subnet.

Implementation:

Configure multipoint subinterfaces on subnet 234.0/24 between routers R2, R3 and R4. Use full mesh of PVCs:

R2:

```
interface Serial0/0
 no ip address
 encapsulation frame-relay IETF
 no frame-relay inverse-arp
 frame-relay lmi-type ansi
!
interface Serial0/0.234 multipoint
```

```
 ip address 172.16.234.2 255.255.255.0
 frame-relay map ip 172.16.234.2 203
 frame-relay map ip 172.16.234.3 203 broadcast
 frame-relay map ip 172.16.234.4 204 broadcast
 no frame-relay inverse-arp
```

R3:

```
interface Serial0/0
 no ip address
 encapsulation frame-relay IETF
 no frame-relay inverse-arp
 frame-relay lmi-type ansi
!
interface Serial0/0.234 multipoint
 ip address 172.16.234.2 255.255.255.0
 frame-relay map ip 172.16.234.2 203
 frame-relay map ip 172.16.234.3 203 broadcast
 frame-relay map ip 172.16.234.4 204 broadcast
 no frame-relay inverse-arp
```

R4:

```
interface Serial0/0
 no ip address
 encapsulation frame-relay IETF
 no fair-queue
 no frame-relay inverse-arp
 frame-relay lmi-type ansi
!
interface Serial0/0.234 multipoint
 ip address 172.16.234.4 255.255.255.0
 frame-relay map ip 172.16.234.2 402 broadcast
 frame-relay map ip 172.16.234.3 403 broadcast
 frame-relay map ip 172.16.234.4 402
 no frame-relay inverse-arp
```

Verification:

Issue the **show frame-relay map** command to verify DLCIs mapping:

```
R2#sh frame-relay map
Serial0/0.234 (up): ip 172.16.234.2 dlci 203(0xCB,0x30B0), static,
              IETF, status defined, active
Serial0/0.234 (up): ip 172.16.234.3 dlci 203(0xCB,0x30B0), static,
              broadcast,
              IETF, status defined, active
Serial0/0.234 (up): ip 172.16.234.4 dlci 204(0xCC,0x30C0), static,
              broadcast,
              IETF, status defined, active
Serial1/0 (up): ip 172.16.25.2 dlci 500(0x1F4,0x7C40), static,
              CISCO, status defined, active
Serial1/0 (up): ip 172.16.25.5 dlci 500(0x1F4,0x7C40), static,
              broadcast,
              CISCO, status defined, active
R2#
```

Issue: Configure Frame Relay physical interfaces on the subnet 172.16.25.0/24.

Solution:

When you examine this link, you will notice that it is not configured through the dedicated Frame-Relay switch. For the dedicated Frame-Relay switch configuration see the beginning of the DOiT workbook. When you examine the dedicated Frame-Relay switch configuration at the beginning of the DOiT workbook, you will notice that only routers R1, R2, R3 and R4 are attached to the dedicated Frame-Relay switch. Router R5 is not connected to the dedicated Frame-Relay switch. This fact forces you to answer the following question: If router R5 is not connected to the dedicated Frame-Relay switch, how can it form a Frame-Relay connection to router R2? The answer to this question is: R5 and R2 can form a direct Frame-Relay connection between each other over a shared synchronous serial connection. A similar configuration requirement was presented in Scenario 3. To configure a direct or “back to back” Frame-Relay connection, perform the following steps:

Configure one of the routers to act as the Frame-Relay switch and the other router to act as the Frame-Relay DTE device. On the router that is configured as the Frame-Relay switch, enter the global configuration command “frame-relay switching” and the interface command “frame-relay intf-type dce” as well as your standard interface command such as “encap frame-relay”. Determine which end is the DCE connection from the synchronous serial interface perspective. You can do this with the following command, “sh controllers serial N (where N is the serial interface number. Do not forget the space between the key word serial and the serial interface number if you want this command to work.) In the **show controllers** display, you will see which interface is the DCE connection and which is the DTE connection. Whichever one is the DCE connection, you must configure the clock rate on that interface. (NOTE: For consistency’s sake, it is recommended to make the side with the DCE connection act as the Frame-Relay switch. It is not required; it is only recommended. The functions performed by a Frame-Relay DCE interface are completely independent of the functions performed by the synchronous serial DCE interface. Again, it is recommended to configure the interface that is the synchronous serial DCE interface as the Frame-Relay DCE interface only for configuration consistency.) Finally, the last thing to remember when configuring a back to back Frame-Relay configuration is: you will not use any “frame-relay route” commands because you are not switching Frame-Relay traffic from one Frame-Relay switch interface to another. Instead of using the “frame-relay route” command, use the “frame-relay interface-dlci XXX” command to advertise a DLCI from the router acting as the Frame-Relay switch to the router acting as the Frame-Relay DTE device. To assure that inverse-arp requirements are satisfied, enter a frame-relay map command on both routers referencing the DLCI announced by the router acting as the Frame-Relay switch. (NOTE: Many people get concerned that if you enter a frame-relay interface-dlci command and a frame-relay map command on the same interface, the frame-relay map command will get erased. This will not happen if you enter the frame-relay interface-dlci command first followed by the frame-relay map command.) This type of configuration is fine when both sides of the configuration possess the same DLCI. In this Scenario both sides in fact use the same DLCI. Therefore, if you perform the configuration steps above, your back-to-back Frame-Relay configuration is complete.

Implementation:

Configure back-to-back Frame Relay connection between R2 and R5:

R2:

```
interface Serial1/0
 ip address 172.16.25.2 255.255.255.0
 ip pim sparse-mode
 encapsulation frame-relay
 ip ospf network point-to-point
 frame-relay map ip 172.16.25.2 500
 frame-relay map ip 172.16.25.5 500 broadcast
 no frame-relay inverse-arp
 frame-relay lmi-type cisco
```

R5:

```
frame-relay switch
!
interface Serial1/0
 ip address 172.16.25.5 255.255.255.0
 ip pim sparse-mode
 encapsulation frame-relay
 ip ospf network point-to-point
 clockrate 72000
 frame-relay map ip 172.16.25.2 500 broadcast
 frame-relay map ip 172.16.25.5 500
 frame-relay interface-dlci 500
 no frame-relay inverse-arp
 frame-relay lmi-type cisco
 frame-relay intf-type dce
```

Verification:

Issue the **show frame-relay map** to verify DLCIs mapping and **show frame-relay pvc** command to verify DLCI status:

```
R5#sh frame map
Serial1/0 (up): ip 172.16.25.2 dlci 500(0x1F4,0x7C40), static,
              broadcast,
              CISCO, status defined, active
Serial1/0 (up): ip 172.16.25.5 dlci 500(0x1F4,0x7C40), static,
              CISCO, status defined, active
R5#

R5#sh frame pvc

PVC Statistics for interface Serial1/0 (Frame Relay DCE)

              Active     Inactive      Deleted       Static
  Local          1            0            0            0
  Switched       0            0            0            0
  Unused         0            0            0            0

DLCI = 500, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial1/0

  input pkts 81354         output pkts 101991       in bytes 5821304
  out bytes 7687755        dropped pkts 0           in pkts dropped 0
  out pkts dropped 0                out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0           out FECN pkts 0
  out BECN pkts 0          in DE pkts 0             out DE pkts 0
  out bcast pkts 47034     out bcast bytes 3616624
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  pvc create time 3d04h, last time pvc status changed 3d04h
R5#
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.2 Catalyst configuration

HIDDEN ISSUES TO SPOT WITH CATALYST CONFIGURATION

Issue: To determine how CAT1 and CAT2 need to be configured, READ AHEAD IN THIS SCENARIO!

Solution:

Read the Catalyst Specialty commands in Section 7.15 of this Scenario. It will provide you with some direction on how to configure VTP and trunk ports. For all other Catalyst configuration requirements, you may apply whatever "best practices" you adhere to.

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.3 Catalyst QoS

HIDDEN ISSUES TO SPOT WITH CATALYST QoSCONFIGURATION

Issue: Port FastEthernet0/10 on CAT1 has Network Engineer's management laptop connected to it. Configure ports connected to VLAN20 to allow Network Engineer to operate on routers R2 and R3 even when network is heavily congested by any type of activity. Use CoS=3 for traffic from management laptop and make sure it has 25% of bandwidth when needed. Other CoS values must have bandwidth according to the table: CoS 6,7 – 20%, CoS 0,1 and 2 – 30%, CoS 4,5 – 25%. Also, traffic from this laptop must always be sent ahead of everything else.

Solution:

Mark traffic arriving on port Fa0/10 with CoS value 3:

```
interface FastEthernet0/10
 switchport access vlan 20
 switchport mode dynamic desirable
 mls qos cos 3
 mls qos cos override
```

Now, set up queueing on both interconnect ports and on destination ports for VLAN20. Example on R3's port on CAT2:

Queues are reassigned, and CoS=3 goes to queue 4, which is made expedited queue. Bandwidth is split according to table: 20%, 30%, 25% and 25% by having wrr-queue bandwidth command on port.

```
interface FastEthernet0/3
 description R1  Fa0/0
 switchport trunk encapsulation isl
 switchport trunk allowed vlan 10,30
 switchport mode trunk
 wrr-queue bandwidth 4 6 5 5
 wrr-queue cos-map 1 6 7
 wrr-queue cos-map 2 0 1 2
 wrr-queue cos-map 3 4 5
 wrr-queue cos-map 4 3
 priority-queue out
```

Verification:

Issue **sh mls qos inte fa0/3 queueing** command (for example):

```
CAT2#sh mls qos inte fa0/3 queueing
FastEthernet0/3
Egress expedite queue: ena
wrr bandwidth weights:
qid-weights
  1 - 4
  2 - 6
  3 - 5
  4 - 5     when expedite queue is disabled (but it's enabled, therefore 4 – expedited)
Cos-queue map:
cos-qid
  0 - 2
  1 - 2
  2 - 2
  3 - 4
  4 - 3
  5 - 3
  6 - 1
  7 – 1
CAT2#
```

Attention

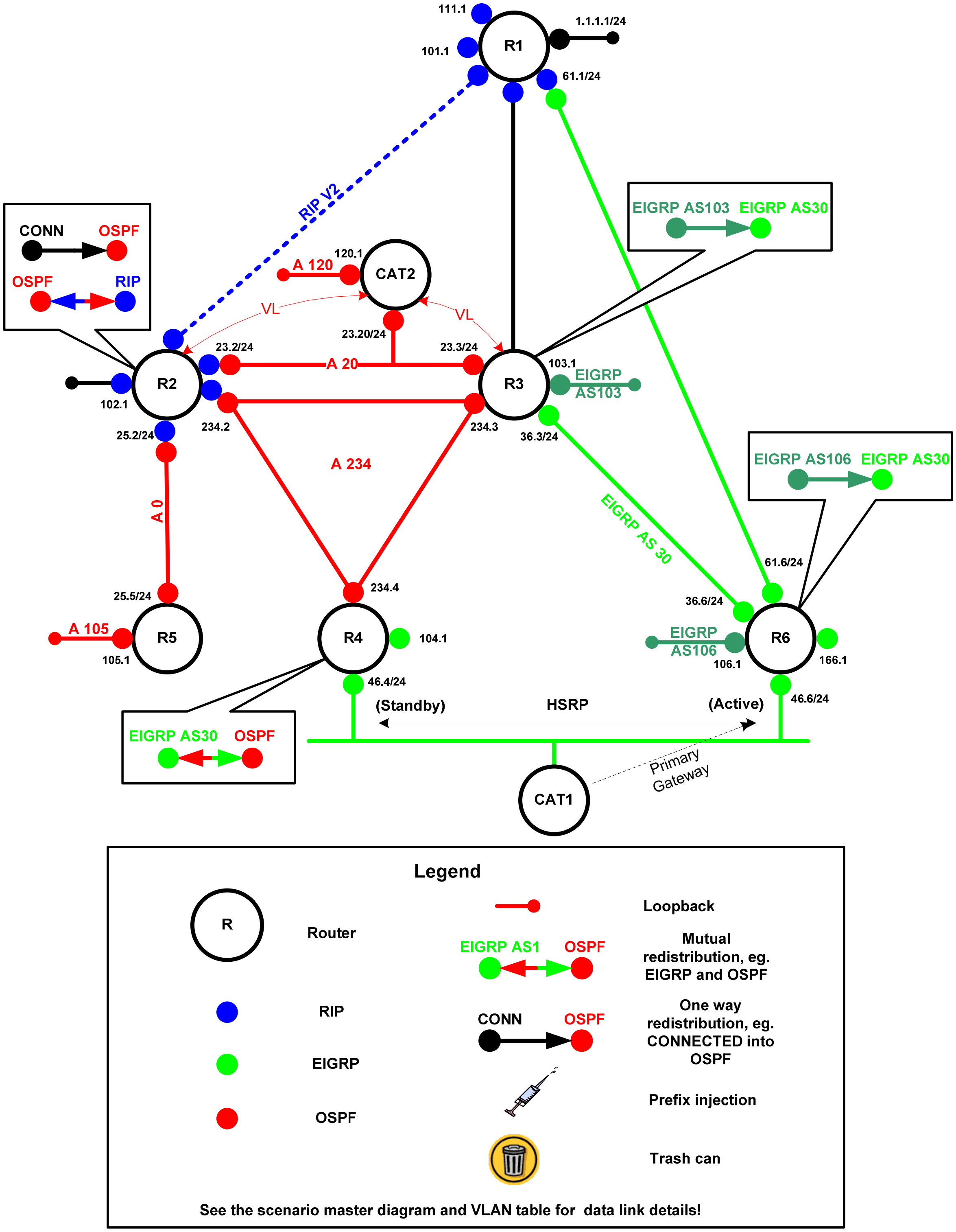
To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

HIDDEN ISSUES TO SPOT WITH ROUTE REDISTRIBUTION

Before examining the specific issues related to configuring each of the IGP's involved in this Scenario, let's survey the entire topology and determine how all of the different IGP's will interoperate. Performing such a survey will force us to consider the issues related to route redistribution.

When evaluating a single internetwork topology that contains multiple routing protocols, a good starting point of analysis is to determine whether there is more than one direct or indirect connecting point between two routing protocols. If there is only one connecting point between two routing protocols, providing connectivity between them is relatively simple. If there are two or more connecting points, then providing connectivity between the two routing protocols can be complex. When two or more connecting points exist, you can use them to provide redundancy as well as load balancing and optimum path selection. However, when two or more connecting points exist, you must also assure, at the very least, that no routing loops exist and, whenever possible, no suboptimal paths are selected.

NOTE: The colors used in this diagram greatly add to the understanding of redistribution applied to this Scenario. If you print this document, attempt to print this page with a color printer.



See the scenario master diagram and VLAN table for data link details!

Redistribution Table

The following table provides a useful summary of which prefixes are imported into a given routing protocol. Pay special attention to the color coding of the table. The colors exactly match the colors used in the diagram. Please make note of the following notation used in the redistribution table:

AD = A manually set “administrative distance”value.
M = A manually set “metric” value.
Protocol(Protocol-1,Protocol-2,etc.) When a protocol is followed by additional protocols enclosed within parentheses, this represents that the first protocol in the list is a transit routing domain for the protocols listed within the parentheses. When the protocol listed first gets redistributed into the target routing domain, it is also redistributes prefixes from the routing protocols listed inside the parenthesis.

Redist Point	Into RIP		Into OSPF		Into EIGRP	
	PERMIT	DENY	PERMIT	DENY	PERMIT	DENY
R2	OSPF		RIPv2 Connected			
R3			EIGRP AS 103 EIGRP AS 30		EIGRP AS 103 EIGRP AS 30 OSPF -> EIGRP 30	
R4			EIGRP 30		OSPF	
R6					EIGRP AS 106 => EIGRP AS 30	

NOTE: The colors used in this table greatly add to the understanding of redistribution applied to this Scenario. If you print this document, attempt to print this page with a color printer.

Key points associated with the redistribution operations performed in this Scenario is provided below.

Redistribution in this scenario is minimal and only provides reachability without redundancy or load balancing. RIP and OSPF are redistributed on R2. OSPF and EIGRP are redistribute on R4. EIGRP interdomain information is redistributed on R3 and R6 and only one way – from local processes into process 30. There are no multiple point redistribution and no loops therefore all redistributions are mutual.

One way to test that your redistribution satisfies the goal of universal connectivity is to run a TCL script like the one below on each router. TCL scripting support is available in the IOS versions used here on routers R1, R2, R5 and R6 (the 3600 models). The simple script below lists all of the IP addresses in our pod. It can be built once in notepad, and then pasted into each router to automate pings. There is a paper on TCL scripting available in the READiT section of the Netmasterclass website. Some addresses are used in later tasks and may not be reachable at this point. Run tclsh in privileged mode, paste the script below, and then issue the command tclq.

```
foreach address {
172.16.101.1
172.16.102.1
172.16.103.1
```

```
        172.16.104.1
        172.16.105.1
        172.16.106.1
        172.16.110.1
        172.16.111.1
        172.16.120.1
        172.16.13.1
        172.16.13.3
        172.16.166.1
        172.16.21.1
        172.16.21.2
        172.16.23.2
        172.16.23.20
        172.16.23.3
        172.16.234.2
        172.16.234.3
        172.16.234.4
        172.16.25.2
        172.16.25.5
        172.16.31.1
        172.16.31.3
        172.16.36.3
        172.16.36.6
        172.16.46.10
        172.16.46.4
        172.16.46.6
        172.16.61.1
        172.16.61.6
        } {ping $address}
```

We also need to make sure that our solution is a stable one. If we have split-horizon or other route feedback problems routes may continually be inserted and removed from our routing tables. We can test stability by observing the output of debug IP routing. Finally, we need to make sure that our routes are optimal: that native prefixes are routed by native protocols and that we are using the shortest paths. This requires close examination of each routing table.

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.4 OSPF

HIDDEN ISSUES TO SPOT WITH OSPF

Issue: CAT2 has no connection to Area 0 and it is connected to 2 different areas.

Solution:

Configure a virtual-link to make the 172.16.120.0/24 subnetwork reachable via Area 0. Make OSPF Area 20 the transit area.

Implementation:

Configure a virtual-link between R2 and CAT2 over area 20:

R2:

```
router ospf 1
 log-adjacency-changes
 area 20 virtual-link 172.16.120.1
```

CAT2:

```
router ospf 1
 log-adjacency-changes
 area 20 virtual-link 172.16.102.1
```

Verification:

Verify that virtual link is up and running by issuing **show ip ospf virtual-links** command:

```
CAT2#sh ip ospf virtual-links
Virtual Link OSPF_VL0 to router 200.200.200.2 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 20, via interface Vlan20, Cost of using 1
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:09
    Adjacency State FULL (Hello suppressed)
    Index 1/3, retransmission queue length 0, number of retransmission 1
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

Issue: Deliver area 0 to R3. Do not use "network … area" command.

Implementation:

Virtual link is also part of area 0. To bring area 0 to R3 – establish virtual link between R2 and R3.

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.5 RIP

HIDDEN ISSUES TO SPOT WITH RIP

Issue: Advertise RIP routing updates on the VLAN 21 and 22 only.

Solution:

Configure “passive-interface default” under the RIP routing process; however, for the Vlan 21 and 22 interfaces enter the command “no passive-interface fa0/0.21” and the same for fa0/0.22.

Implementation:

R1:

```
router rip
 version 2
 passive-interface default
 no passive-interface FastEthernet0/0.21
 no passive-interface FastEthernet0/0.22
 network 172.16.0.0
 no auto-summary
```

R2:

```
router rip
 version 2
 passive-interface default
 no passive-interface FastEthernet0/0.21
 no passive-interface FastEthernet0/0.22
 network 172.16.0.0
 no auto-summary
```

Verification:

Issue the **show ip protocol | be “rip”** command to see which interfaces are sending/receiving RIP updates:

```
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 27 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: ospf 1, rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0.21    2     2
    FastEthernet0/0.22    2     2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
  Passive Interface(s):
    FastEthernet0/0
    Serial0/0
    Serial0/0.234
```

Issue: Accept RIP updates only if they pass clear text authentication.

Solution:

Configure clear text authentication for RIP and apply it to the VLAN 21 and 22 interfaces. RIP authentication keys are defined in global configuration mode and applied on a specific interface.

Implementation:

Configure a key chain and apply it to the interfaces involved in RIP routing:

R1:

```
key chain RIP
 key 1
  key-string exam7
!
interface FastEthernet0/0.21
 ip address 172.16.21.1 255.255.255.0
 ip rip authentication key-chain RIP
!
interface FastEthernet0/0.22
 ip address 172.16.22.1 255.255.255.0
 ip rip authentication key-chain RIP
```

R2:

```
key chain RIP
 key 1
  key-string exam7
!
interface FastEthernet0/0.21
 ip address 172.16.21.2 255.255.255.0
 ip rip authentication key-chain RIP
!
interface FastEthernet0/0.22
 ip address 172.16.22.2 255.255.255.0
 ip rip authentication key-chain RIP
```

Verification:

Issue the **show ip protocol | inc “rip”** command:

```
R2#sh ip proto | be "rip"
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 22 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: ospf 1, rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0.21    2     2                    RIP
    FastEthernet0/0.22    2     2                    RIP
R2#
```

7.6 EIGRP

HIDDEN ISSUES TO SPOT WITH EIGRP

Issue: There are four routers running EIGRP in this Scenario; however, there are three EIGRP Autonomous Systems.

Solution:

In order for all EIGRP routes to get propagated, you must manually redistribute between two EIGRP processes. On router R3, you must redistribute between EIGRP AS 103 and 30. On router R6, you must redistribute between EIGRP AS 103 and 106.

Implementation:

Redistribute between EIGRP processes on R3:

```
router eigrp 30
 redistribute eigrp 103 metric 10000 100 255 1 1500
```

Redistribute between EIGRP processes on R6:

```
router eigrp 30
 redistribute eigrp 106 metric 10000 100 255 1 1500
```

Verification:

To verify that redistribution is working, issue the **show ip eigrp topology** command:

```
R3#sh ip eigrp topo
IP-EIGRP Topology Table for AS(30)/ID(172.16.103.1)

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

[skipped]

P 172.16.103.0/24, 1 successors, FD is 128256
         via Redistributed (128256/0)
IP-EIGRP Topology Table for AS(103)/ID(172.16.103.1)

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

P 172.16.103.0/24, 1 successors, FD is 128256
         via Connected, Loopback103
R3#
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.7 BGP

HIDDEN ISSUES TO SPOT WITH BGP

Issue: Configure AS 100 on R1 and AS 300 on R3. Form a BGP peer relationship between them over VLAN 10.

Solution:

The issue with this task is that two IP addresses are assigned to both R1’s and R3’s VLAN 10 attached interfaces. On R1, the primary IP address on its F0/0 interface is 172.16.31.1 and its secondary is 172.16.13.1. R3 possesses the same addressing scheme but in reverse. A neighbor relationship can be formed with this addressing arrangement using the primary IP address assigned to the respective remote-peer. In addition to following this requirement, an EBGP multihop commands needs to be configured on both routers R1 and R3.

IOS will setup bgp session from secondary address if remote peer is on secondary subnet. To work around this issue, use **update-source** modifier, which will get primary IP address of the interface.

Implementation:

R1:

```
interface FastEthernet0/0
 ip address 172.16.13.1 255.255.255.0 secondary
 ip address 172.16.31.1 255.255.255.0
router bgp 100
 no synchronization
 network 172.16.13.0 mask 255.255.255.0
 network 172.16.31.0 mask 255.255.255.0
 neighbor 172.16.13.3 remote-as 300
 neighbor 172.16.13.3 ebgp-multihop 2
```

R3:

```
interface FastEthernet0/0.10
 encapsulation isl 10
 ip address 172.16.31.3 255.255.255.0 secondary
 ip address 172.16.13.3 255.255.255.0
router bgp 300
 network 172.16.13.0 mask 255.255.255.0
 network 172.16.31.0 mask 255.255.255.0
 neighbor 172.16.31.1 remote-as 100
 neighbor 172.16.31.1 ebgp-multihop 2
 neighbor 172.16.31.1 update-source FastEthernet0/0.10
```

Verification:

Issue the **show ip bgp summary** command to verify that bgp peering is up:

```
R1#sh ip bgp sum
BGP router identifier 172.16.111.1, local AS number 100
BGP table version is 15, main routing table version 15
6 network entries using 606 bytes of memory
[skipped]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.13.3     4   300    4804    4804       15    0    0 01:33:23            3
R1#
```

Issue: Advertise the VLAN 21 and 22 subnets 172.16.21.0/24 and 172.16.22.0/24. Make sure AS 100 does not destabilize the BGP domain if the these links go up and down.

Solution:

The challenge to this task is to understand how a subnet that is going up and down could destabilize a BGP domain. When the link goes up, routes associated with that link will be added to the routing table. When the link goes down, routes associated with that link may be removed. If these associated prefixes are injected into BGP, they could cause a BGP route flap condition. You can prevent the insertion and deletion of the individual subnets by aggregating the 172.16.21.0/24 and 172.16.22.0/24 prefixes in the following manner: “aggregate-address 172.16.20.0 255.255.252.0”.

A second method of solving the flapping IP address generated by the PPP “peer neighbor-route” command is to configure the following command within the router bgp configuration mode “bgp dampening”. When the “bgp dampening” command is entered, prefixes that are continually advertised in BGP updates and subsequently withdrawn will be “dampened”. If the “bgp dampening” solution is applied to this Scenario, the command will be placed on the routers receiving BGP updates from router R1. In the supplied topology, the “bgp dampening” command will be placed on router R3. If router R3 receives a combination of BGP updates and withdraws for a specific prefix – in this case the PPP “peer neighbor-route” address – it will get dampened. To configure “bgp dampening” for a specific prefix, you can associate a route-map with the command. Furthermore, you can manipulate “bgp dampening” options within a route-map with the “set dampening” command. Although configuring the “bgp dampening” is a viable solution to the stated configuration task, the Scenario Seven final configuration script applies the BGP aggregation solution described earlier.

Implementation:

```
router bgp 100
 template peer-session INTER-AS
  transport connection-mode passive
  description Session to AS 300
  ebgp-multihop 2
 exit-peer-session
 !
 no synchronization
 bgp log-neighbor-changes
 network 1.1.1.0 mask 255.255.255.0
 network 172.16.13.0 mask 255.255.255.0
 network 172.16.31.0 mask 255.255.255.0
 aggregate-address 172.16.20.0 255.255.252.0
 redistribute connected metric 1 route-map Connected-To-BGP
 neighbor 172.16.13.3 remote-as 300
 neighbor 172.16.13.3 inherit peer-session INTER-AS
 no auto-summary
```

Issue: Make sure all routers within AS300 exchange NLRI with a local preference of 200. Do not use a route-map to accomplish this.

Solution:

If you can't set the local-preference for a set of prefixes with a route-map command, enter the following command under the bgp routing process on routers R2 and R5: "**bgp default local-preference 200**". This will set the local-preference to 200 for all bgp prefixes received.

Verification:

Issue the **show ip bgp** command and verify that default local preference is assigned and it is 200:

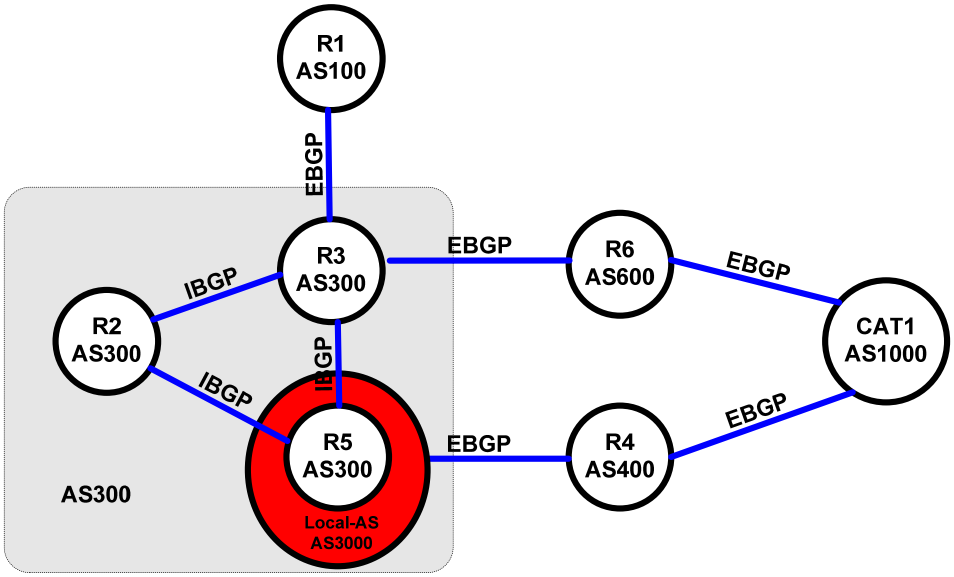
```
R2#sh ip bgp
BGP table version is 4209, local router ID is 172.16.102.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i1.1.1.0/24       172.16.31.1              0    200      0 100 i
*>i10.10.10.0/24    172.16.36.6              0    200      0 600 1000 i
r>i172.16.13.0/24   172.16.23.3              0    200      0 i
*>i172.16.20.0/22   172.16.31.1              0    200      0 100 i
r>i172.16.31.0/24   172.16.23.3              0    200      0 i
R2#
```

Issue: Configure AS 3000 on router R5.

Solution:

R5 has already been assigned to AS 300. Even though R5 has been assigned to AS 300, it can also "appear" to be assigned to a second AS to specific EBGP neighbors using the following BGP command: "**neighbor X.X.X.X local-as**". The "neighbor /local-as" command can only be applied to EBGP neighbor relationships. Since this Scenario directs you to configure one and only EBGP neighbor relationship on R5 with R5 appearing as if it is in AS 3000, the "neighbor/local-as" command fulfills the configuration requirement. All other R5 neighbor relationships in this Scenario are IBGP neighbor relationships. With these neighbor relationships, R5 should appear as a member of AS 300. To better understand how its neighbors perceive R5, see the following diagram:



Implementation:

Configure neighbor relationship between R5 and R4 using local-as keyword:

```
router bgp 300
 no synchronization
 neighbor 172.16.234.4 remote-as 400
 neighbor 172.16.234.4 local-as 3000
 neighbor 172.16.234.4 ebgp-multihop 2
```

Verification:

Verify that R5 appears as AS 3000 on R4 by issuing the **show ip bgp summary** command on R4:

```
R4#sh ip bgp summary
BGP router identifier 172.16.104.1, local AS number 400
[skipped]

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.25.5     4  3000     101      98        6    0    0 01:33:15        5
172.16.46.10    4  1000     104     100        6    0    0 01:33:52        5
R4#
```

Issue: Peer AS 3000 and AS 400 between R4 and R5.

Solution:

Since routers R4 and R5 do not possess a common subnet and since they are EBGP peers, you need to configure ebgp-multihop to successfully form the EBGP neighbor relationship between them.

Implementation:

```
      router bgp 300
       no synchronization
       neighbor 172.16.234.4 remote-as 400
       neighbor 172.16.234.4 local-as 3000
       neighbor 172.16.234.4 ebgp-multihop 2
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.8 BGP Part 2

HIDDEN ISSUES TO SPOT WITH BGP FEATURES

Issue: Make sure that ebgp-multihop is not present in neighbor command between R1 and R3 in BGP configuration on R1. Also, configure R1 neighbor as passive-only in transport for BGP. Make sure the same set of arguments is not configured twice would another similar link between R1 and AS300 appear later on.

Solution:

Configure session template:

```
      router bgp 100
       template peer-session INTER-AS
        transport connection-mode passive
        description Session to AS 300
        ebgp-multihop 2
       exit-peer-session
       !
       no synchronization
       bgp log-neighbor-changes
       network 1.1.1.0 mask 255.255.255.0
       network 172.16.13.0 mask 255.255.255.0
       network 172.16.31.0 mask 255.255.255.0
       aggregate-address 172.16.20.0 255.255.252.0
       redistribute connected metric 1 route-map Connected-To-BGP
       neighbor 172.16.13.3 remote-as 300
       neighbor 172.16.13.3 inherit peer-session INTER-AS
       no auto-summary
```

Transport setting will make sure R1 will never originate this session. However, BGP router with highest RID will originate session in peers. Therefore, BGP RID on R3 must be increased to successfully set up BGP peering.

Ebgp-multihop parameter is now in template and not on neighbor statement. Also, new neighbors can inherit these settings too without specifying them twice or more.

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.9 IOS Features

HIDDEN ISSUES TO SPOT WITH IOS FEATURES

Issue: Configure R5 so that you can reboot the router with an "snmpset" UNIX command.

Solution:

In order to fulfill this requirement, enter the following two commands on router R5 in global configuration mode: (1) snmp-server system-shutdown and (2) snmp-server community NMC RW.

Implementation:

```
snmp-server engineID local 00000009020000107B81C363
snmp-server community private RW
snmp-server system-shutdown
snmp-server enable traps tty
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.10 Router Maintenance

HIDDEN ISSUES TO SPOT WITH ROUTER MAINTENANCE

Issue: Configure at least 40 small buffers and no more than 170 big buffers on router R1 .

Solution:

System buffer pools contain packets that are "process" switched (as opposed to fast cache or CEF switched). The IOS creates different sets of buffers used to hold packets of different sizes. IOS buffer sizes include: small (104 bytes), medium (600 bytes), big (1524 bytes), VeryBig (4520 bytes), Large (5024 bytes), and Huge (18024 bytes). You can set buffer parameters such as the maximum and/or minimum number of a specific buffer type in global configuration mode. Commands that can be used to tune the system buffers are buffers <type: small/large/etc> permanent xxx, buffers <type: small/large/etc> max-free xxx, buffers <type: small/large/etc min-free xxx where xxx is the number of buffers.

Implementation:

Configure buffers using the following commands:

```
buffers small min-free 40
buffers small initial 40
buffers big max-free 170
```

Verification:

To verify buffers settings, issue **show buffers** command:

```
R1#sh buffers
Buffer elements:
     500 in free list (500 max allowed)
     448698 hits, 0 misses, 0 created

Public buffer pools:
Small buffers, 104 bytes (total 50, permanent 50, peak 90 @ 3d08h):
     46 in free list (40 min, 150 max allowed)
     312813 hits, 0 misses, 40 trims, 40 created
     0 failures (0 no memory)
Middle buffers, 600 bytes (total 25, permanent 25, peak 43 @ 3d08h):
     25 in free list (10 min, 150 max allowed)
     11943 hits, 6 misses, 18 trims, 18 created
     0 failures (0 no memory)
Big buffers, 1536 bytes (total 50, permanent 50):
     50 in free list (5 min, 170 max allowed)
     19434 hits, 0 misses, 0 trims, 0 created
     0 failures (0 no memory)
R1#
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.11 Security

HIDDEN ISSUES TO SPOT WITH SECURITY

Issue: Do not use a standard or extended numbered access-list to accomplish this task .

Solution:

Instead of using a NUMBERED access-list to accomplish this task, use a named access-list. Some entries in the named access-list that you create are:

```
deny   icmp any any redirect (for ICMP redirects)
deny   ip any host 255.255.255.255 (for local broadcasts)
deny   ip any host 0.0.0.0 (for 0.0.0.0/0 routes)
deny   ip any 224.0.0.0 15.255.255.255 (for multicasts)
deny   ip 127.0.0.0 0.255.255.255 any (the reserved loopback addr block)
deny   ip 10.0.0.0 0.255.255.255 any (RFC 1918 reserved addresses)
deny   ip 172.16.0.0 0.15.255.255 any (RFC 1918 reserved addresses)
deny   ip 192.168.0.0 0.0.255.255 any (RFC 1918 reserved addresses)
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.12 IPv6

HIDDEN ISSUES TO SPOT WITH IPv6

Issue: Configure IPv6 on all interfaces marked with IPv6 addresses on IPv6 diagram.

Solution:

IPv6 address must be configured in interface configuration mode using the **ipv6 address** command:

```
interface FastEthernet0/0
 ip address 172.16.13.1 255.255.255.0 secondary
 ip address 172.16.31.1 255.255.255.0
 duplex auto
 speed auto
 ipv6 address FEC0::13:1/125
 ipv6 rip RIPv6 enable
```

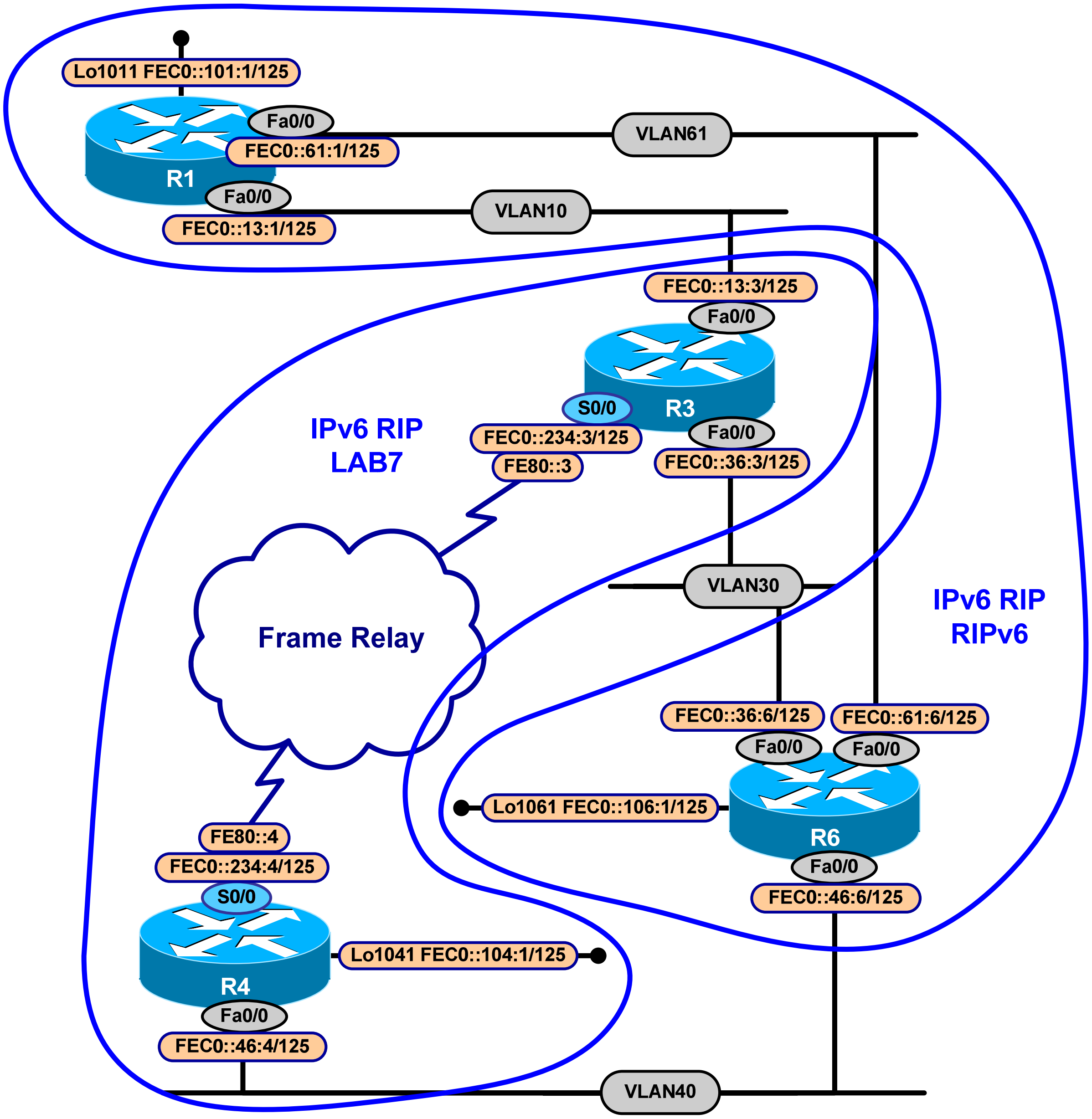
Verification:

Verify that interface picked up IPv6 configuration, and that the peer address is reachable:

```
R1#ping fec0::13:3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FEC0::13:3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R1#sh ipv6 inte fa0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::2D0:58FF:FE95:C8A1
  Global unicast address(es):
    FEC0::13:1, subnet is FEC0::13:0/125
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:FF13:1
    FF02::1:FF95:C8A1
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
R1#
```



Issue: Configure RIPng on all IPv6 links.

Solution:

To enable RIPng on interfaces use **ipv6 rip *rip-process-name* enable**. Remember, that rip-process-name is locally significant on router, and different names will not prevent routing information exchange between different routers.

```
interface FastEthernet0/0.61
 encapsulation dot1Q 61
 ip address 172.16.61.6 255.255.255.0
 no snmp trap link-status
 ipv6 address FEC0::61:6/125
 ipv6 rip RIPv6 enable
```

Verification:

Check that RIPng has next-hop neighbors and how many prefixes are received from those neighbors:

```
R6#sh ipv6 rip next-hops
 RIP process "RIPv6", Next Hops
  FE80::2D0:58FF:FE95:C8A1/FastEthernet0/0.61 [3 paths]
  FE80::204:C1FF:FE8E:19E0/FastEthernet0/0.30 [3 paths]
  FE80::250:54FF:FE7C:A640/FastEthernet0/0.40 [3 paths]
```

Issue: Make R3 send packets towards R4's loopback FEC0::104:1 in 2:1 ratio over FastEthernet and Serial link in favor of FastEthernet. Do not create any new addresses. Do not use commands that have word "unnumbered".

Solution:

RIPng doesn't allow unequal path load balancing. The trick in this question is to create tunnel over Ethernet between R3 and R4, enable IPv6 on that tunnel, set it's type to ipv6ip and enable RIP over it.

Then, RIP needs to be tuned on R3 that metrics of FEC0::104:1 are equal from all three interfaces using metric-offset keyword. At the same time, R4 must not balance anything, so metric of prefixes coming thru the tunnel must be increased in favor of other interfaces.

R3:

```
interface Tunnel20
 no ip address
 ipv6 enable
 ipv6 rip LAB7 enable
 ipv6 rip LAB7 metric-offset 2
 tunnel source 172.16.36.3
 tunnel destination 172.16.46.4
 tunnel mode ipv6ip

interface Serial0/0.234 multipoint
ipv6 rip LAB7 metric-offset 2

ipv6 router rip LAB7
 maximum-paths 3
```

R4:

```
interface Tunnel20
 no ip address
 ipv6 enable
 ipv6 rip LAB7 enable
 ipv6 rip LAB7 metric-offset 2
 tunnel source 172.16.46.4
 tunnel destination 172.16.36.3
 tunnel mode ipv6ip
```

Verification:

Verify that R3 has 3 routes installed for FEC0::104:0/125 and 2 out of 3 will be routed over Ethernet either natively or encapsulated:

```
R3#sh ipv6 route fec0::104:0
IPv6 Routing Table - 13 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
R   FEC0::104:0/125 [120/3]
     via FE80::4, Serial0/0.234
     via FE80::AC10:2E04, Tunnel20
     via FE80::202:4BFF:FE19:9280, FastEthernet0/0.30
R3#
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.13 QoS

HIDDEN ISSUES TO SPOT WITH QoS

Issue: Configure WFQ on the Frame-Relay links between R2 and R5 with the listed parameters.

Solution:

Weighted Fair Queuing (WFQ) is enabled by default on Cisco synchronous serial WAN interfaces operating at 2.048 Mbps rates and lower. You can verify this by entering "show interface Serial X" (where X is the interface number). You will see the default WFQ settings on the Serial interface. Cisco maintains two implementations of WFQ: platform independent WFQ and Distributed WFQ (DWFQ is available on VIP interfaces). We are going to discuss only the platform independent WFQ. Furthermore, there are two variations of the Cisco platform independent WFQ implementation: Flow-based WFQ and Class Based WFQ. This task addresses the configuration of flow based WFQ. Class Based WFQ will be encountered in other Scenarios in this workbook. As its name reflects, flow based WFQ is designed to "fairly" distribute interface resources during times of congestion in a manner so that no traffic gets "unfairly" starved out. This is done on a per-flow basis. When you hear the term "flow" used in an IP QOS context, it means that traffic is classified by the following characteristics: source and destination IP address, IP protocol number, source and destination TCP or UDP port numbers. Flow-based WFQ uses these packet characteristics – as well as the IP precedence bit settings - to classify traffic. WFQ automatically and dynamically classifies traffic into different "dynamic" queues based upon criteria of packets received such as: precedence bits settings of an IP packet, source and destination IP address, IP protocol type, as well as source and destination UDP/TCP port numbers. The IP precedence setting of a packet possesses particular importance with flow-based WFQ. All active WFQ flows sharing the same IP precedence setting are allocated that same interface bandwidth when WFQ is active. Furthermore, active WFQ flows with a higher precedence setting get a greater share of the interface bandwidth. Once the traffic is classified using the parameters mentioned above, it is assigned to a “dynamic queue”. The default number of dynamic queues is 256. The number of dynamic queues can be adjusted with the “fair-queue” interface configuration command. As incoming packets are classified and assigned to a “dynamic queue”, they are assigned weights. The weights are mapped to a WFQ sequence number which in turn determines when a specific packet will be transmitted out a WFQ enabled interface. Although the operation of flow-based WFQ is automatic, you can modify its configuration with the fair-queue interface configuration command. The fair-queue interface configuration command has three configuration parameters: (1) congestive-discard-threshold (2) dynamic-queues and (3) reservable-queues. The congestive-discard-threshold sets a threshold that determines when to drop packets. When you adjust the congestive-discard-threshold, you must adjust it in units that are in powers of 2 ranging from 16 to 4096 (16, 32, 64,128,etc.) The default setting congestive-discard-threshold is 64. The dynamic queues parameter establishes the number of queues used for “best-effort” flows requiring no optimized treatment. When you adjust the number of dynamic queues you must do so using at a minimum the following values : 16, 32, 64, 128,256, 512,1024, 2048 and 4096 Note that all of these values are powers of 2. Reservable queues are allocated for WFQ preferential traffic. You can assign a value of 0 to 1000 for reservable queues. There is no “power of 2” requirement. The default reservable queue setting is 0. Other IOS QOS features such as RSVP use reservable queues. Finally, when you apply the fair-queue command, you must apply it on a physical interface. You cannot apply the fair-queue command on a subinterface. To fulfill the Flow-based WFQ requirements of this task, enter the following command under the appropriate Serial interface on routers R2 and R5: “fair-queue 128 512 10”. Remember Class Based-WFQ scenarios will appear in other Scenarios in this workbook.

Implementation:

R2:

```
interface Serial1/0
 fair-queue 128 512 10
```

R5:

```
interface Serial1/0
 fair-queue 128 512 10
```

Verification:

Issue **show queueing fair** command to verify fair queue settings:

```
R5#sh queueing fair
Current fair queue configuration:

  Interface           Discard    Dynamic  Reserved  Link    Priority
                      threshold  queues   queues    queues  queues
  Serial1/0           128        512      10        8       1
  Serial1/1           64         256      0         8       1
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as “show all”.

7.14 Catalyst Specialties

HIDDEN ISSUES TO SPOT WITH CATALYSE SPECIALTIES

Issue: Configure VTP version 2 on CAT2. CAT1 will be a server of VTP domain "NMC" with the password "NMC" as well.

Solution:

You can set the VTP version used on a Catalyst 3550 with the following global configuration command: "vtp version X" (X= version 1 or 2). Configure the password "NMC" on both CAT1 and CAT2 to provide VTP level password protection.

Implementation:

Configure VTP server on CAT1:

```
vtp mode server
vtp domain NMC
vtp password NMC
vtp version 2
```

Configure VTP client on CAT2:

```
vtp mode client
vtp domain NMC
vtp password NMC
```

Make sure that client device doesn't have VLAN.DAT with vlan information on it's flash, since this will prevent VTP domain from correct synchronization.

Verification:

Verify VTP status by issuing **show vtp status** command:

```
CAT1#sh vtp sta
VTP Version                     : 2
Configuration Revision          : 1
Maximum VLANs supported locally : 1005
Number of existing VLANs        : 12
VTP Operating Mode              : Server
VTP Domain Name                 : NMC
VTP Pruning Mode                : Disabled
VTP V2 Mode                     : Enabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0xC8 0x65 0x13 0xF9 0x7F 0x61 0xC8 0x91
Configuration last modified by 172.16.46.10 at 3-1-93 11:42:27
Local updater ID is 172.16.46.10 on interface Vl40 (lowest numbered VLAN interface found)

CAT2#sh vtp status
VTP Version                     : 2
Configuration Revision          : 1
```

```
Maximum VLANs supported locally : 1005
Number of existing VLANs        : 12
VTP Operating Mode              : Client
VTP Domain Name                 : NMC
VTP Pruning Mode                : Disabled
VTP V2 Mode                     : Enabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0xC8 0x65 0x13 0xF9 0x7F 0x61 0xC8 0x91
Configuration last modified by 172.16.46.10 at 3-1-93 11:42:27
```

Verify VLAN configuration by issuing **show vlan brief** on VTP client switch:

```
CAT2#sh vlan brie | inc active
1    default                          active    Fa0/2, Fa0/4, Fa0/6, Fa0/7
10   VLAN0010                         active
20   VLAN0020                         active
21   VLAN0021                         active
22   VLAN0022                         active
30   VLAN0030                         active
40   VLAN0040                         active    Fa0/5
61   VLAN0061                         active
```

Issue: Restrict the amount of flooded traffic on the trunks.

Solution:

Limit the amount of VLANs that can traverse all trunks configured on both CAT1 and CAT2 to only the VLANs used in this Scenario. You can do this by configuring the following interface configuration command on each trunk port: “switchport trunk allow vlans xxx” where xxx is one or more vlans to be allowed on the trunk port. If multiple vlan’s are listed separate them with a comma or hyphen (the hyphen denotes a range of vlan’s) with no spaces. An alternative to restricting the level of flooding performed on trunk ports can be attained by enabling vtp pruning. This can be performed by entering the following global configuration mode command: “vtp pruning”.

Implementation:

Issue the **switchport trunk allow vlans 10,20-22,30,40,61** command on interfaces Fa0/13, Fa0/14 and Po1 of CAT1 and CAT2.
Issue the **switchport trunk allow vlans 10,30** command on interface Fa0/3 of CAT2.
Issue the **switchport trunk allow vlans 10,20-22,61** command on interface Fa0/1 of CAT2.
Issue the **switchport trunk allow vlans 30,40,61** command on interface Fa0/8 of CAT1.
Issue the **switchport trunk allow vlans 20-22** command on interface Fa0/2 of CAT1.

Verification:

Issue the **show interface trunk** command to verify trunking interfaces:

```
CAT1#sh inte trunk

Port        Mode         Encapsulation  Status        Native vlan
Fa0/2       on           802.1q         trunking      20
Fa0/8       on           802.1q         trunking      1
Fa0/24      desirable    n-isl          trunking      1
Po1         on           802.1q         trunking      1
```

```
Port      Vlans allowed on trunk
Fa0/2       20-22
Fa0/8       30,40,61
Fa0/24      1-4094
Po1         10,20-22,30,40,61

Port        Vlans allowed and active in management domain
Fa0/2       20-22
Fa0/8       30,40,61
Fa0/24      1,10,20-22,30,40,61
Po1         10,20-22,30,40,61

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/2       20-22
Fa0/8       30,40,61
Fa0/24      1,10,20-22,30,40,61

Port        Vlans in spanning tree forwarding state and not pruned
Po1         10,20-22,30,40,61
```

Attention!

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.15 Address Administrator

HIDDEN ISSUES TO SPOT WITH ADDRESS ADMINISTRATION

Issue: Router CAT1 should prefer R4 as a primary gateway based on a Cisco developed protocol. In case of R4 failure, CAT1 should prefer R6.

Solution:

The "Cisco developed protocol" referenced in this task that allows a device to select a primary gateway is the Hot Standby Routing Protocol (HSRP). This is an HSRP configuration task without ever explicitly mentioning HSRP. In order to fulfill the specific requirements of this Section, configure HSRP on the Ethernet interface of R4 and the FastEthernet interface of R6. R4 should be assigned a higher HSRP priority (lower decimal value) than R6 since it is supposed to be the primary gateway for VLAN 40.

Implementation:

Configure HSRP between R4 and R6:

R4:

```
interface Fa0/4
 standby ip 172.16.46.1
 standby priority 110
 standby preempt
```

R6:

```
interface FastEthernet0/0.40
 encapsulation dot1Q 40
 standby ip 172.16.46.1
 standby preempt
```

Verification:

To verify HSRP configuration issue the **show standby** command:

```
R6#sh standby
FastEthernet0/0.40 - Group 0
  State is Standby
    13 state changes, last state change 04:53:11
  Virtual IP address is 172.16.46.1
  Active virtual MAC address is 0000.0c07.ac00
    Local virtual MAC address is 0000.0c07.ac00 (default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.344 secs
  Preemption enabled
  Active router is 172.16.46.4, priority 110 (expires in 9.828 sec)
  Standby router is local
  Priority 100 (default 100)
  IP redundancy name is "hsrp-Fa0/0.40-0" (default)
R6#
```

Issue: Be able to ping the rest of the network from 172.16.110.1 from CAT1.

Solution:

Configure an ip local policy on CAT1 and assign the IP address used by the HSRP configuration described above as the default-gateway address. You accomplish this be entering the following command under the route-map configured for the local policy: "set **ip default next-hop 172.16.46.1**". A local policy configuration is activated in global configuration mode. It applies to all traffic that originates from the router that is configured with the local policy command.

Implementation:

```
ip local policy route-map Default-Route

route-map Default-Route permit 10
 set ip default next-hop 172.16.46.1
```

To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".

7.16 Multicast

HIDDEN ISSUES TO SPOT WITH MULTICAST

Issue: Configure a PIM Sparse Mode network. Do not use the Cisco proprietary method of distributing RP information throughout the multicast network.

Solution:

The Cisco proprietary method of distributing RP information throughout a multicast network is “Auto-RP”. Since you are restricted from using “Auto-RP”, you must distribute RP information by configuring either the bootstrap routing protocol or via multiple static “ip pim rp-address” configuration statements. With this configuration only PIM sparse mode needs to be configured on the interfaces participating in multicasting. Task 7.17.2 mentions that you must make CAT2 the root of the shared tree for multicast groups 228.8.8.8 and 227.7.7. If CAT2 must be the root of the shared tree for these multicast groups, it must be configured as the Rendezvous Point (RP) for these multicast groups. This is performed by entering the following global configuration command: “ip pim rp-candidate loopback 0” The interface that you select must have PIM configured on it and it must be reachable by all multicast routers. Not only must you configure an RP candidate RP, you must also configure a “bootstrap router”. This is performed with the following global configuration command: “ip pim bsr loopback 0”.

Implementation:

Enable multicast routing on all routers, participating in multicast by issuing the **ip multicast-routing** command. Enable PIM on all interfaces participating in multicast routing by issuing the **ip pim sparse-mode** command.

Configure CAT2 as bsr-candidate and rp-candidate:

```
ip pim bsr-candidate Loopback0 0
ip pim rp-candidate Loopback0 group-list Multicast-RP

ip access-list standard Multicast-RP
 permit 227.7.7.7
 permit 228.8.8.8
```

Join the loopbacks to the multicast group 227.7.7.7 by issuing the **ip igmp join 227.7.7.7** command. Also, enable PIM on these loopbacks as well, otherwise they will not return ICMP echo.

Verification:

To verify group-to-RP mapping, issue the **show ip pim rp map** command:

```
R2#sh ip pim rp map
PIM Group-to-RP Mappings

Group(s) 227.7.7.7/32
  RP 172.16.120.1 (?), v2
    Info source: 172.16.120.1 (?), via bootstrap, priority 0
         Uptime: 04:58:46, expires: 00:01:36
```

```
Group(s) 228.8.8.8/32
  RP 172.16.120.1 (?), v2
    Info source: 172.16.120.1 (?), via bootstrap, priority 0
         Uptime: 04:58:46, expires: 00:01:37
R2#
```

Issue: Configure each member router to process only Join and Prune messages destined for 172.16.120.1 and 227.7.7.7 group while building the shared tree.

Solution:

In order to fulfill this requirement, use the global configuration command “ip pim accept-rp x.x.x.x yy” command (x.x.x.x is the specific rp to access; yy is the access-list associating the rp with a specific multicast address). For this Scenario, this command will be entered in the following manner: ip pim accept-rp 172.16.120.1 99 (172.16.120.1 is an IP address assigned to a loopback interface on the Rendezvous Point CAT2. 99 is an access-list that possesses one entry 227.7.7.7, the multicast group that CAT2 is acting as the Rendezvous Point for. Note: This requirement is not applied to JOIN and PRUNE messages generated during the SPT cutover.

Implementation:

```
ip pim accept-rp 172.16.120.1 Accept-RP
ip access-list standard Accept-RP
 permit 227.7.7.7
```

Issue: ***Generate (172.16.31.1, 227.7.7.7) traffic.***

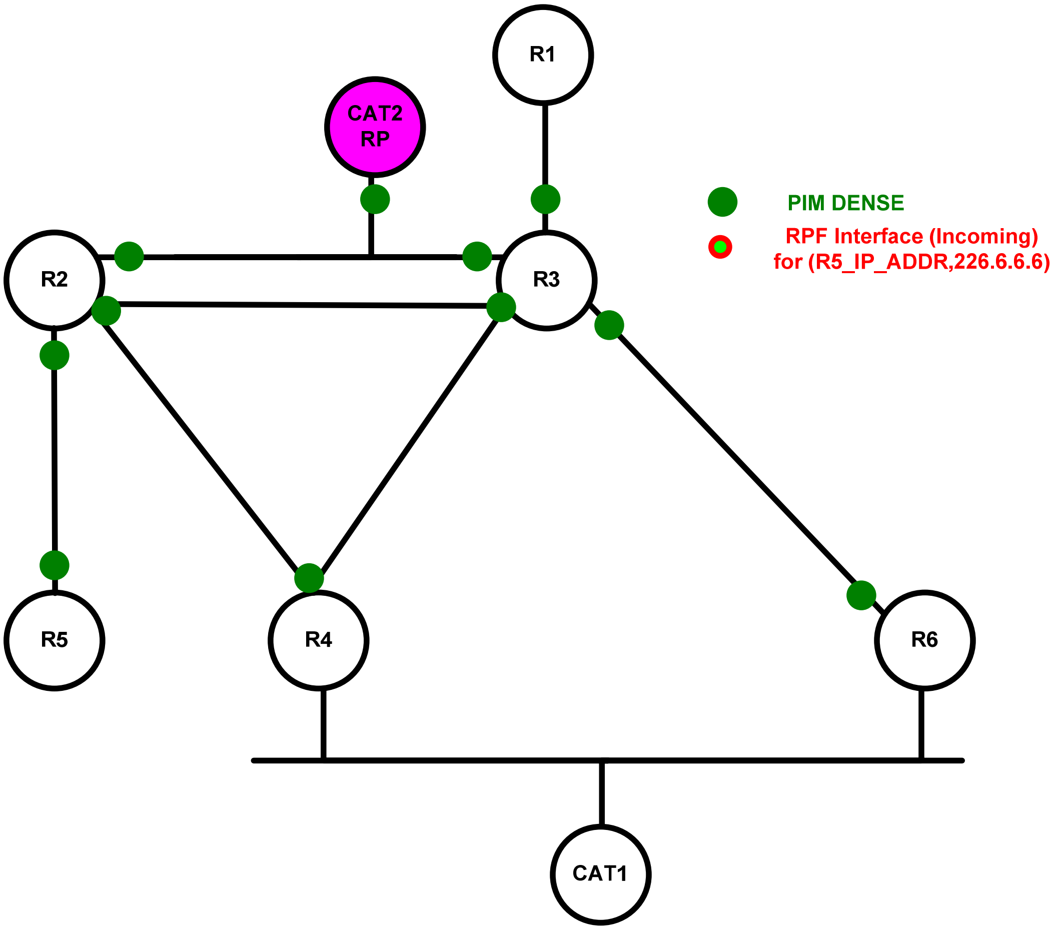
Solution:

The 172.16.31.1 IP address is assigned to the FastEthernet interface of R1. Therefore, you must originate your multicast test traffic from R1. If you want to control precisely what interfaces multicast traffic exist or what the source IP address is assigned to a given stream of multicast traffic, use an extended PING. You will find options to provide such control. This IP address was never originated via any IGP in this Scenario. However, it was originated by BGP. Therefore, it should be reachable by all participating multicast receivers.

Implementation:

Issue the **ping 227.7.7.7** command to initiate multicast traffic.

The following diagram illustrates the final configuration to fulfill the requirements of this Section.



To obtain a comprehensive view of the configuration tasks in this section, access the SHOWiT engine. With the SHOWiT engine, you can enter in over 1000 IOS commands as well a collection of NMC proprietary commands such as "show all".