CCIE Enterprise Infrastructure Foundation

Rough Cuts

ciscopress.com

Narbik Kocharians, CCIE® No. 12410

CCIE Enterprise Infrastructure Foundation

Narbik Kocharians

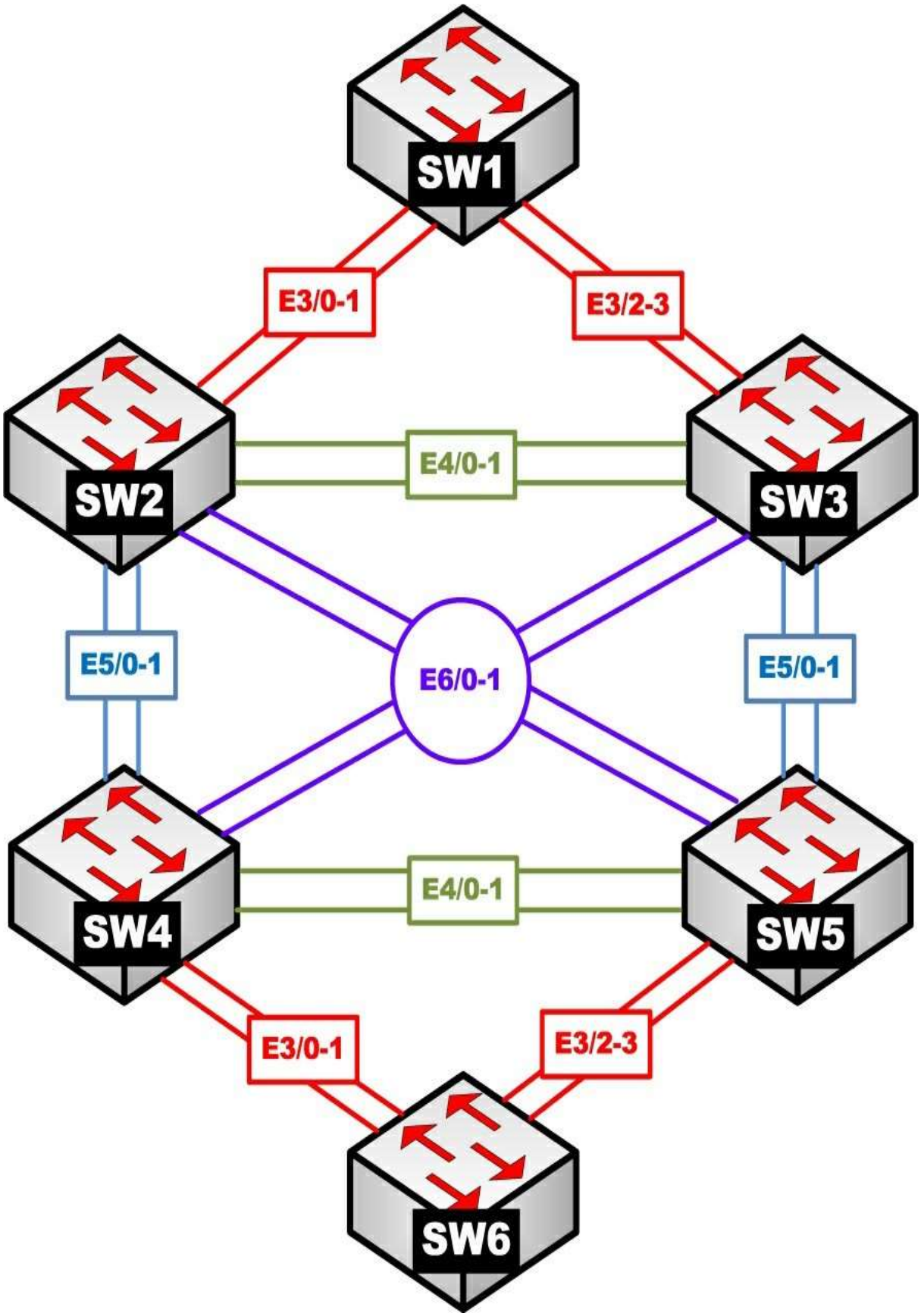
Pearson IT Certification

Contents

Table of Contents

Chapter 1. Switching

Lab 1: Configuring Trunks


SW1
E3/0-1
E3/2-3
SW2
E4/0-1
SW3
E5/0-1
E6/0-1
E5/0-1
SW4
E4/0-1
SW5
E3/0-1
E3/2-3
SW6

What is covered in this lab:

This lab focuses mainly on configuring trunk links and VLANs and controlling which VLANs are allowed on a particular trunk link. This lab covers the following topics: dynamic trunks, basic VTP operation, how to modify the allowed VLAN lists on trunk links, and VTP pruning.

This lab should be conducted on the enterprise POD.

Task 1

Shut down all ports on all six switches and configure the VTP domain name to be **TST**.

Task 2

Configure the following hostnames:

Second switch: SW2

Third switch: SW3

Fourth switch: SW4

Fifth switch: SW5

Task 3

Configure a dot1q trunk between SW3 and SW5 using their E0/5 interfaces, based on the following policy:

SW3, E5/0

This port should be configured into a permanent trunk mode, and it should negotiate to convert the neighboring interface into a trunk.

SW5, E5/0

This port should be configured to actively attempt to convert the link to a trunk. You should not configure **switchport trunk encapsulation dot1q** on this port.

Task 4

Configure a trunk between SW3 and SW5, using their E5/1 interfaces. You should use an industry-standard protocol for the trunk encapsulation, based on the following policy:

SW3, E5/1

This port should be configured into permanent trunk mode, and it should negotiate to convert the neighboring interface into a trunk.

SW5, E5/1

This port should be configured to negotiate a trunk only if it receives negotiation packets from a neighboring port; this port should never start the negotiation process. You should not configure **switchport trunk encapsulation dot1q** on this port.

Task 5

Configure a trunk link between SW4 and SW5, using their E4/0 interfaces. These ports should be configured to negotiate the neighboring interface into a dot1q trunk, but they should not be in permanent trunk mode.

Task 6

Configure a dot1q trunk between SW4 and SW5, using their E4/1 interfaces, based on the following policy:

SW4, E4/1

This port should be configured to actively attempt to convert the link to a trunk. This port should not be in permanent trunking mode.

SW5, E4/1

This port should be configured to negotiate a trunk only if it receives negotiation packets from a neighboring port; this port should never start the negotiation process or be configured with the **switchport trunk encapsulation dot1q** command.

Task 7

Configure a dot1q trunk between SW3 and SW4, using their E6/0 interfaces; these switches should be configured into permanent trunk mode and negotiate the neighboring interface into a trunk.

Task 8

Configure a dot1q trunk between SW3 and SW4, using their E6/1 interfaces. These ports should not use DTP to negotiate a trunk.

Task 9

Configure trunking on the E4/0-1 interfaces of SW2 and SW3, the E5/0-1 interfaces on SW2 and SW4, and the E6/0-1 interfaces of SW2 and SW5. These ports should be in permanent trunk mode.

Task 10

Configure the following VLANs on SW2 and ensure that they are propagated to the other switches:

VLANs 2–10, 100, 200, 300, 400, 230, 350, 450, 240, 250, and 340

Task 11

Configure the trunks based on the following policy:

Policy Item	Trunk Interface	Between Switches	Allowed VLAN/s
1	E4/1	SW2 ←→ SW3	Only 230
2	E5/0	SW3 ←→ SW5	Only 350
3	E4/0	SW4 ←→ SW5	Only 450
4	E5/0	SW2 ←→ SW4	Only 240
5	E6/0	SW2 ←→ SW5	Only 250
6	E6/0	SW3 ←→ SW4	Only 340

Task 12

Add VLANs to the allowed list of the trunks, based on the following chart:

Policy Item	Trunk Interface	Between Switches	Add to the Allowed VLAN/s
1	E4/1	SW2 ←→ SW3	100
2	E5/0	SW3 ←→ SW5	200
3	E4/0	SW4 ←→ SW5	300
4	E6/0	SW2 ←→ SW5	400

Task 13

Remove VLANs from the allowed list of trunks, based on the following chart:

Policy Item	Trunk Interface	Between Switches	Allowed VLAN/s
1	E5/1	SW2 ←→ SW4	Remove 1, 4 – 10 only
2	E5/1	SW3 ←→ SW5	Remove 2, 4 – 10 only

Task 14

Configure SW2, SW3, and SW5, based on the following chart:

Policy Item	Trunk Interface	Between Switches	Allowed VLAN/s
1	E4/0	SW2 ←→ SW3	None
2	E6/1	SW2 ←→ SW5	None

Task 15

Configure SW2, SW4, and SW5, based on the following chart:

Policy Item	Trunk Interface	Between Switches	Allowed VLAN/s
1	E4/1	SW4 ←→ SW5	All except 450
2	E5/1	SW2 ←→ SW4	All except 240

Task 16

Configure SW3 and SW4, based on the following chart. You may override some of the previous tasks to accomplish this task.

Policy Item	Trunk Interface	Between Switches	Allowed VLAN/s
1	E6/0	SW3 ←→ SW4	All
2	E6/1	SW3 ←→ SW4	All

Task 17

Erase the config.text and vlan.dat files on SW1-5 and reload them before proceeding to the next task.

Task 18

Configure SW2 and SW3 based on the following policies:

- Configure the hostnames of Switch 2 and Switch 3 to be SW2 and SW3, respectively.
- Shut down all the ports on SW2 and SW3.
- Configure a dot1q trunk between SW2 and SW3 using port E4/0.
- Ensure that both switches belong to the VTP domain TST.

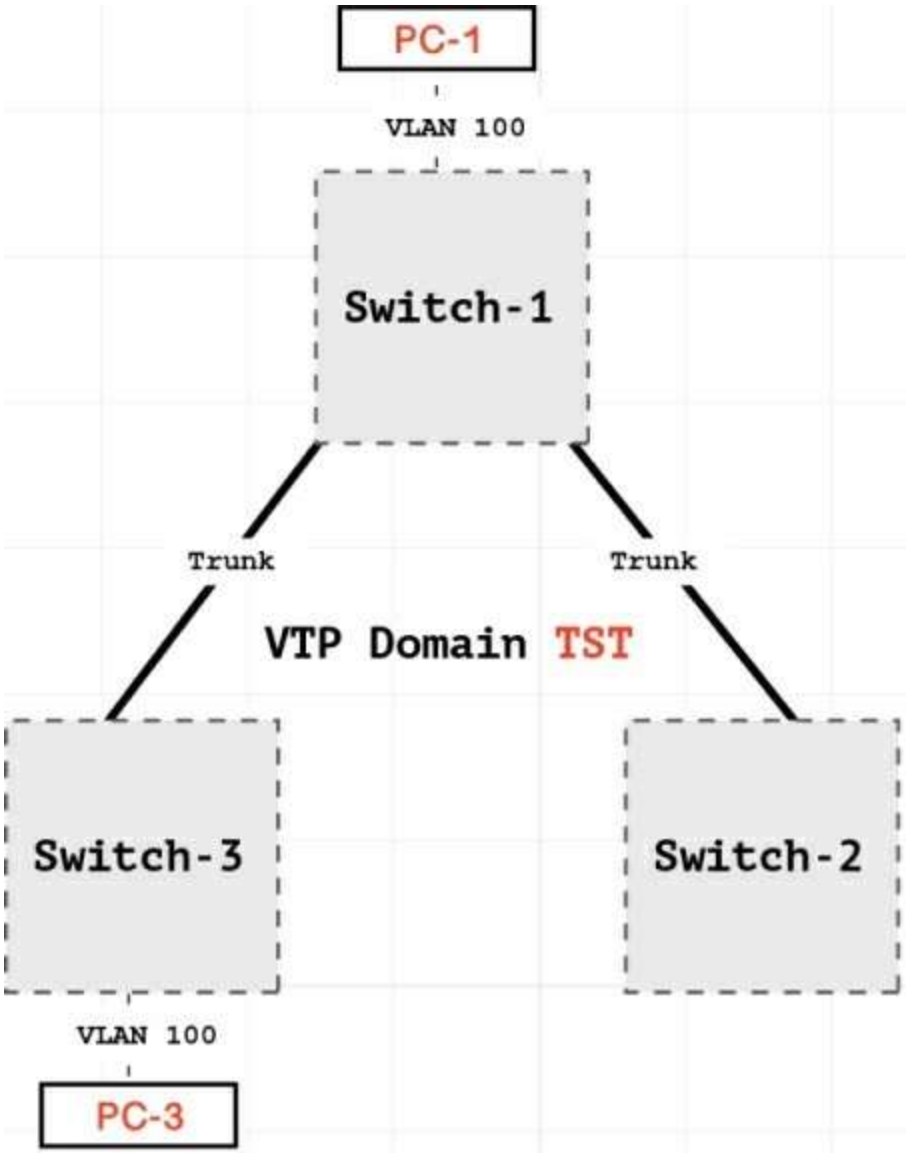
Task 19

Configure VLAN 100 on SW2 and assign its E0/1 interface to this VLAN.

VTP Pruning

In Task 10, it was explained that VTP is a protocol that can be used to synchronize the VLAN databases between switches that participate in the same VTP domain. VTP uses three message types to achieve this synchronization: **summary advertisement**, **advertisement request**, and **subset advertisements**. This synchronization process makes it such that an administrator only needs to configure VLANs on a single VTP server switch and have those configurations synchronized to all remaining switches in the VTP domain.

VTP can be used for more than simple VLAN database synchronization. It can be used to reduce unnecessary broadcast flooded traffic originating in a specific VLAN from entering sections of the network that do not require that VLAN traffic. For example, the sample topology from Task 10 is examined:



In this topology, the interfaces connecting Switch-1 to Switch-2 and Switch-3 are configured as trunk links. All three switches belong to the same VTP domain, **TST.**

Previously, VLAN 100 was created on Switch-1. Switch-1 replicated this VLAN to Switch-2 and Switch-3 using VTP. Looking at the diagram, however, only Switch-1 and Switch-3 have interfaces connected to hosts in VLAN 100. Switch-2 does not have any interfaces connected to VLAN 100. Since Switch-2 has no interfaces in VLAN 100, there is no need for it to receive broadcast traffic originating from hosts in VLAN 100 connected to other switches in the network.

When PC-1 needs to reach PC-3 in the topology, it starts by sending a broadcast ARP frame to learn PC-3's MAC address. The following packet capture shows that the ARP frame was broadcasted to both Switch-2 and Switch-3:

On Switch-3:

```
Frame 4990: 64 bytes on wire (512 bits), 64
bytes captured (512 bits) on interface 0
Ethernet II, Src: aa:bb:cc:00:04:20
(aa:bb:cc:00:04:20), Dst: Broadcast
(ff:ff:ff:ff:ff:ff)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 100
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)      Sender MAC
address: aa:bb:cc:00:04:20
(aa:bb:cc:00:04:20)
    Sender IP address: 100.1.1.1
Target MAC address: 00:00:00_00:00:00
(00:00:00:00:00:00)
```

```
    Target IP address: 100.1.1.2
```

On Switch-2

```
Frame 4134: 64 bytes on wire (512 bits), 64
bytes captured (512 bits) on interface 0
Ethernet II, Src: aa:bb:cc:00:04:20
(aa:bb:cc:00:04:20), Dst: aa:bb:cc:00:05:10
(aa:bb:cc:00:05:10)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 100
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)     Sender MAC
address: aa:bb:cc:00:04:20
(aa:bb:cc:00:04:20)
    Sender IP address: 100.1.1.1
Target MAC address: aa:bb:cc:00:05:10
(aa:bb:cc:00:05:10)
    Target IP address: 100.1.1.2
```

Switch-2 has received the broadcast traffic even though it will ultimately drop it because it does not have any hosts in that VLAN. The VLAN 100 broadcast in this case was unnecessarily broadcasted to Switch-2. This unnecessary broadcast occurred because Switch-1 included VLAN 100 in its allowed VLAN list for the trunk link connected to Switch-2, as shown here:

```
Switch-1#show interfaces trunk

Port        Mode             Encapsulation
Status        Native vlan Et0/0
on               802.1q
trunking      1 Et0/1       on
802.1q         trunking      1

Port        Vlans allowed on trunk
Et0/0       1-4094
Et0/1       1-4094

Port        Vlans allowed and active
in management domain Et0/0
1,100,200
Et0/1       1,100,200

Port        Vlans in spanning tree forwarding
state and not pruned Et0/0       1,100 Et0/1
```

```
1,100 ! VLAN 100 is allowed on the trunk link
to Switch-2
```

This inclusion extends VLAN 100's broadcast domain to Switch-2. One way to stop the unnecessary broadcast of VLAN 100 traffic to Switch-2 is to manually remove VLAN 100 from the Switch-1/Switch-2 trunk link by using the **switchport trunk allowed-vlan remove 100** command in interface configuration mode. This process, called *manual pruning*, is demonstrated extensively in previous tasks.

The problem with manual pruning is that if a host were connected to Switch2 in VLAN 100, that VLAN would need to be manually added to the allowed VLAN list on the Switch-1/Switch-2 trunk link again. Multiply this process by hundreds of VLANs and potentially hundreds of clients moving around in the network, and it is clear that this manual pruning process can cause high administrative overhead.

This is where **VTP pruning** comes into play. VTP pruning is a feature of VTP that allows dynamic pruning of VLANs from trunk links where the VLAN traffic is not needed. The process utilizes a fourth VTP message type, called the **VTP membership advertisement**, or **VMA**. These are basically VTP Join/Prune messages.

The basic process for VTP pruning is that a switch sends a VMA for all VLANs for which it is interested in receiving traffic. The switch makes this determination based on whether or not it contains any interfaces that are currently associated with a VLAN that exists in the local switch's VLAN database.

Once again, referring back to the sample topology shown earlier, when VTP pruning is enabled on all the switches, they exchange VMAs with each other. Because Switch-3 is interested in receiving traffic for VLAN 100, its VMA includes VLAN 1 and VLAN 100, as shown in the following capture.

Note

VLAN 1 is pruning ineligible, which means this VLAN will always be included in the advertised active VLAN field and cannot be removed.

From Switch-3:

```
VLAN Trunking Protocol
    Version: 0x01     Code:
Join/Prune Message (0x04)
    Reserved: 00
    Management Domain Length: 3
    Management Domain: TST
    First VLAN ID: 0
    Last VLAN ID: 1007     Advertised active
(i.e. not pruned) VLANs
        VLAN: 1
        VLAN: 100
```

The **Advertised active (i.e. not pruned) VLANs** field indicates the VLANs for which Switch-3 is interested in receiving traffic. Any VLAN that is not assigned to an interface on Switch-3 is omitted from this list. For example, the VMA sent from Switch-2 to Switch-1 includes only VLAN 1. This is because Switch-2 does not have any interfaces assigned to VLAN 100, and thus it has no need for VLAN 100 traffic. Any traffic sent by Switch-1 to Switch-2 in VLAN 100

would inevitably be a futile effort since Switch-2 would end up dropping the traffic anyway.

From Switch-2

```
VLAN Trunking Protocol
    Version: 0x01     Code:
Join/Prune Message (0x04)
    Reserved: 00
    Management Domain Length: 3
    Management Domain: TST
    First VLAN ID: 0
    Last VLAN ID: 1007     Advertised active
(i.e. not pruned) VLANs
         VLAN: 1
```

The following shows the state of the **show interface trunk** output on Switch1. As a result of the VMA message exchange, Switch-1 will now send traffic for VLAN 100 out its trunk link E0/0 to Switch-3 only.

```
Switch-1#show interfaces trunk

Port        Mode             Encapsulation
Status        Native vlan Et0/0
on               802.1q
trunking      1 Et0/1        on
802.1q         trunking      1

Port        Vlans allowed on trunk
Et0/0       1-4094
Et0/1       1-4094

Port        Vlans allowed and active
in management domain Et0/0
1,100,200
Et0/1       1,100,200

Port        Vlans in spanning tree forwarding
state and not pruned
Et0/0       1,100 ! VLAN 1 and 100 allowed to
```

```
Switch-2 Et0/1        1 ! Only VLAN 1 allowed
to Switch-
2
```

VTP pruning is disabled by default and can be enabled by using the **vtp pruning** command in global configuration mode. VTP keeps a list of VLANs that are allowed to be dynamically pruned; it is called the **pruning-eligible list** or the **pruning VLANs enabled list**, depending on the **show** command being used t. This list can be seen using the **show interface switchport** command, as shown here:

```
Switch-1#show int e0/0 switchport | in Pruning
Pruning VLANs Enabled: 2-1001
```

The pruning-eligible list can be modified much like the allowed VLAN list of a trunk link with the **switchport trunk pruning vlan** command and the following additional parameters:

```
Switch-1(config-if)#switchport trunk pruning
vlan ?   WORD    VLAN IDs of the allowed
VLANs when this port is in trunking mode
add      add VLANs to the current list
except   all VLANs except the following
none     no VLANs

  remove  remove VLANs from the current list
```

The result of this command can be seen with the sample topology shown earlier. Before any configuration changes are made, the **show interface e0/1 pruning** command output on Switch-1 reveals that it is pruning VLAN 100 on the E0/1 trunk link connected to Switch-2. This is a result of the missing VLAN 100 in the VMA sent by Switch-2, as shown here:

```
Switch-1#show interface e0/1 pruning

Port                Vlans pruned for lack of
request by neighbor Et0/1                 100

Port                Vlan traffic requested of
neighbor Et0/1                  1,100
```

Next, VLAN 200 is configured on Switch-2. The pruning-eligible list on Switch-2 is then modified to include only VLAN 200 with the **switchport trunk pruning vlan 200** command:

```
Switch-2(config-if)#vlan 200
Switch-2(config-vlan)#exit
Switch-2(config)#int e0/1 Switch-2(config-
if)#switchport trunk pruning vlan 200
```

Notice how the pruning-eligible list on Switch-2 shows only VLAN 200:

```
Switch-2#show int e0/1 switchport | in Pruning
Pruning VLANs Enabled: 200
```

The result of this configuration is that all other VLANs (except VLAN 1 and VLANs 1002 through 1005) are now considered to be pruning ineligible. As a result, the VMA sent by Switch-2 to Switch-1 will include VLAN 100, thus preventing Switch-1 from pruning this VLAN off the trunk link connected to Switch-2. You can see this in the following packet capture, where VLAN 100 is included in the **Advertised active (i.e. not pruned) VLANs** field:

```
VLAN Trunking Protocol
    Version: 0x01     Code:
Join/Prune Message (0x04)
    Reserved: 00
    Management Domain Length: 3
    Management Domain: TST
    First VLAN ID: 0
    Last VLAN ID: 1007
    Advertised active (i.e. not pruned) VLANs

        VLAN: 1
        VLAN: 100
```

Tasks 20 through 27 pertain to VTP pruning for the lab topology.

Task 20

Configure the switches such that they restrict flooded traffic to those trunk links the traffic must use to access the appropriate network device(s).

Task 21

Configure VLANs 200, 300, 400, 500, and 600 on SW2 and ensure that these VLANs are propagated to SW3.

Task 22

Configure the E3/3 interface of SW3 in VLAN 100.

Task 23

Configure the switches such that only VLAN 300 is pruned.

Task 24

Configure the switches such that VLAN 200 is also pruned. You should not use the command from the previous task to accomplish this task.

Task 25

Configure SW2 and SW3 such that none of the VLANs are pruned.

Task 26

Configure SW2 and SW3 such that all configured VLANs in the VLAN database are pruned.

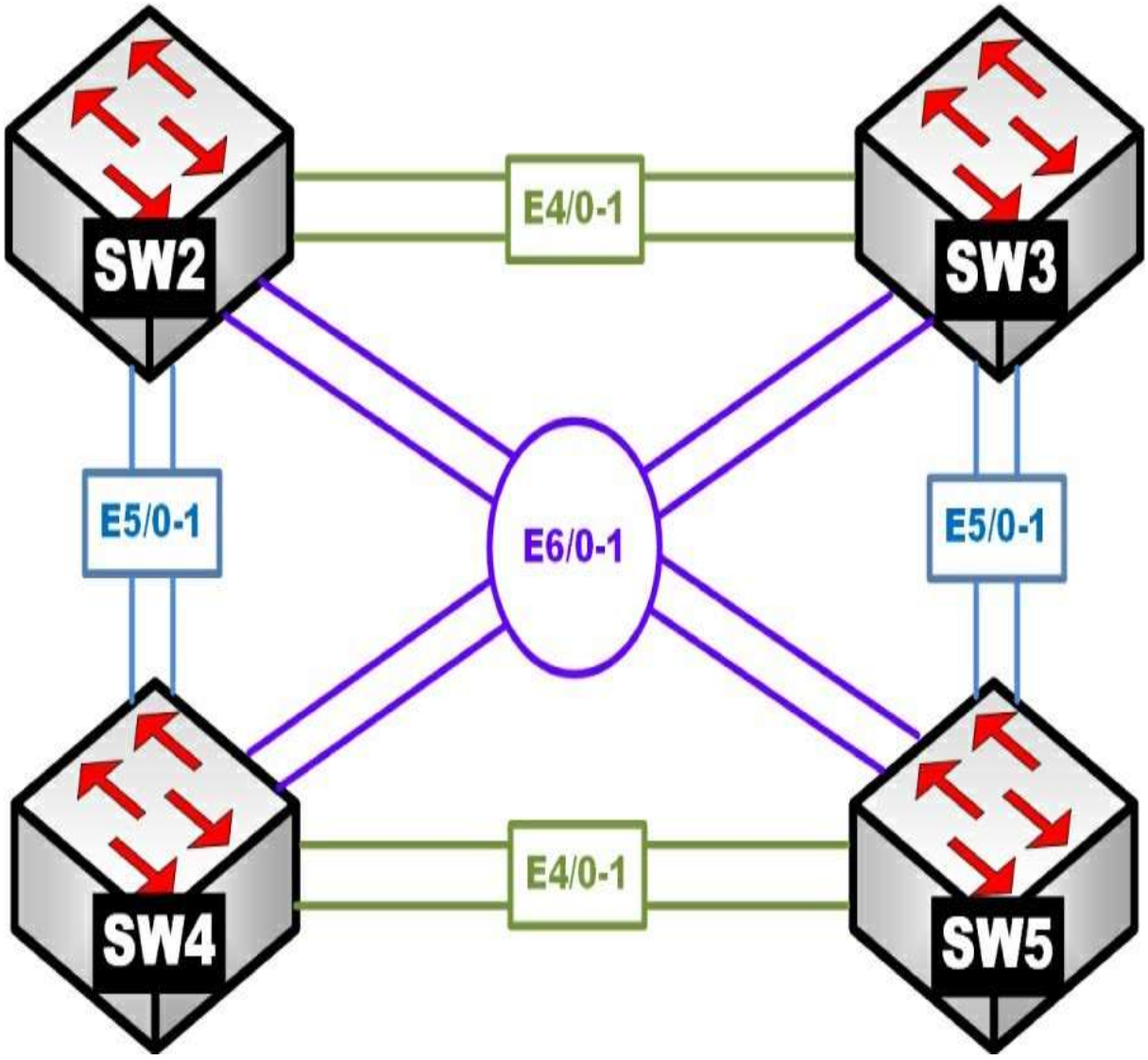
Task 27

Configure SW2 and SW3 such that VLAN 200 is no longer pruned; do not use a command that was used before to accomplish this task.

Task 28

Erase the vlan.dat and config.text files and reload the switches before proceeding to the next lab.

Lab 2: Configuring EtherChannels



This lab should be conducted on the enterprise rack.

Task 1

Configure the hostname of the switches based on the provided diagram. Ensure that all the ports of these four switches are in shutdown mode. Configure these four switches in a VTP domain called **TST**.

Task 2

Configure ports E4/0 and E4/1 on SW2 and SW3 as trunk links, using an industry-standard protocol. These links should appear to Spanning Tree Protocol as a single link. If one of the links fails, the traffic should use the other link, without any interruption. The ports on SW2 should be configured such that they only respond to PAgP packets and never start the negotiation process.

Task 3

Configure ports E5/0 and E5/1 on SW2 and SW4 as trunk links, using an industry-standard protocol. These links should appear to Spanning Tree Protocol as a single link. If one of the links fails, the traffic should use the other link, without any interruption. These ports should *not* negotiate by exchanging **LACP** or **PAgP** protocols to accomplish this task.

Task 4

Ensure that all the EtherChannels created on SW2 are load balanced based on the destination MAC address.

Task 5

Configure ports E6/0 and E6/1 on SW3 and SW4 as a single Layer 3 link; SW3 should be configured with the IP address 34.1.1.3 /24, and SW4 should be configured with the IP address 34.1.1.4 /24. These ports should not negotiate LACP or PAgP.

Task 6

Erase the startup configuration and vlan.dat files before proceeding to the next lab.

Lab 3: Introducing Spanning Tree Protocol

This section is designed to teach basic to advanced concepts of Spanning Tree Protocol-.

It utilizes a common topology over which each version of Spanning Tree Protocol is configured with a given set of requirements and restraints. The requirements and restraints are engineered to explain the behaviors of each version of Spanning Tree Protocol, highlighting the important limitations and enhancements of each feature.

This lab is entirely focused on Spanning Tree Protocol technologies and assumes basic knowledge of the following:

- Trunking configuration
- VLAN configuration and usage
- Layer 2 link aggregation
- IP address configuration
- Basic IP connectivity testing

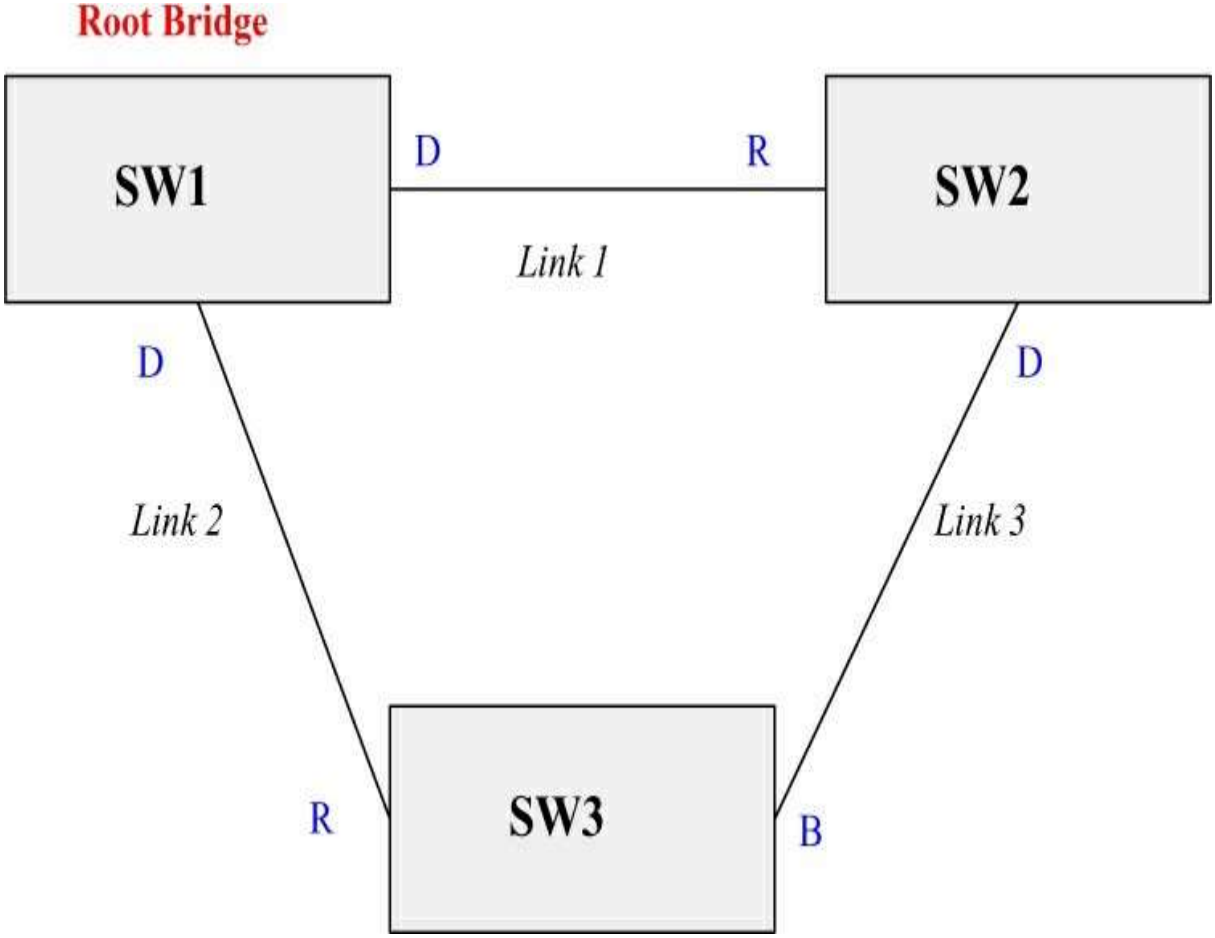
Note

The solutions provided in this lab are not all inclusive. There may be many ways to solve each task. All alternate solutions are acceptable, provided that they do not violate previous restraints or tasks.

802.1D Per-VLAN Spanning Tree Protocol

This lab should be conducted on the mock rack.

Initial Lab Setup



Task 1

Change the hostname on each switch to SW#, where # is the number of the switch (for example, Switch 1 = SW1).

Task 2

Ensure that only the following ports on the switches are in an up/up state:

- SW1
- E2/1-2

- E3/1-2

- SW2
- E1/1-2
- E3/1-2
- E4/1-2
- E5/1-2
- SW3
- E1/1-2
- E2/1-2
- E4/1-2
- E5/1-2
- SW4
- E2/1-2
- E3/1-2
- E6/1-2
- SW5
- E2/1-2
- E3/1-2
- E6/1-2
- SW6
- E4/1-2
- E5/1-2

Task 3

Configure VLANs 10, 20, 30, and 40 on each switch, using any method.

Task 4

Configure trunk ports on all up/up interfaces.

Configuration Tasks: 802.1D

Before getting into the tasks, it helps to review the basic operation of Spanning Tree Protocol and some of the terms that will be used throughout the solution guide. These concepts will be fleshed out more throughout the guide.

802.1D Spanning Tree Protocol is the protocol that is run between switches used to prevent Layer 2 bridging loops. Loops can form in Layer 2 networks because the Layer 2 Ethernet frame does not contain any maximum hop count limitation, such as the TTL field in the IP header. This omission of maximum hop count means a single Ethernet frame can be forwarded infinitely throughout a switched network. This is particularly dangerous when the frame being forwarded is a broadcast frame.

Switches are called *transparent bridges* because they abstract the physical network design from the end stations connecting to the LAN. In the past, all stations connected to a LAN were physically connected to the same physical electrical bus via a repeater. Only one station could speak on the segment at a time, and all stations were considered directly connected. This setup was called a *single collision domain* because there was a possibility that two stations would transmit at the same time, and those transmissions could collide.

Bridges were then created to allow intelligent learning. An Ethernet bridge would separate the LAN into two separate collision domains. Bridges learned MAC addresses and forwarded traffic between network segments only when necessary. This operation was transparent to the end stations because as far as the stations could detect, they were directly connected to stations they were communicating with.

An Ethernet switch is basically a multiport bridge. Instead of learning MAC addresses only on two ports, a switch learns MAC addresses on all ports, allowing microsegmentation of the LAN segment. Because not all traffic forwarded across the LAN segment is repeated to all ports, the switch

achieves a single collision domain between each port on the switch and the stations connected to those ports. For the switch to do this, it must first learn which MAC addresses are reachable from its various ports. To do so, the switch reads the source MAC address of every frame it sends and records it in a MAC address table, also called a CAM table. The switch switches traffic between ports by performing a lookup in the CAM table, based on the destination MAC address of the Ethernet frames. Known unicast traffic can therefore be forwarded only out the port where the intended station exists.

A problem arises when a switch receives a frame destined to a MAC address it has not learned. The switch cannot drop the frame because that would break the LAN communication. Instead, it floods the frame out all ports in the same VLAN on the switch (including trunk ports that carry those VLANs) except the port on which it initially received the frame. This process, known as **unknown unicast flooding**, can cause a loop condition in the switched network with broadcast frames.

Broadcast frames are frames that are intended to be received by all stations. They are sent to the destination MAC address FFFF.FFFF.FFFF. When a switch encounters a broadcast frame, it performs unknown-unicast-type flooding for the frame in question. It sends the traffic out all interfaces except the interface on which it was received.

If there are redundant links in a multi-switch environment, where the chain of interconnected links leads back to the switch that originally forwarded the broadcast traffic, a **broadcast storm** occurs. In a broadcast storm, each receiving switch performs the same unknown-unicast-type flooding on the broadcast packet. The broadcast packet is therefore regenerated and looped endlessly throughout the switched network. This leads to high CPU utilization on the switches and can quickly bring down the entire L2 network.

Spanning Tree Protocol converts a switch network into a shortest-path tree. This tree is constructed by designating a switch as the root of the tree, called the **root bridge**. The root bridge is the only bridge in which all of the ports are considered designated ports. A designated port is a port that is responsible for relaying spanning-tree-related messages downstream from the root bridge to other leaf switches. From the root bridge's perspective, all other

switches are downstream from it, and thus it should be designated on all of its ports.

All non-root switches elect a single port to become their root port. The root port is the port that the switch uses to reach the root bridge. This port also receives spanning-tree-related messages on the shortest-path tree to be relayed out all other designated ports on the switch.

Finally, redundant links in the network are put into a blocking state. So-called blocking ports are ports that receive BPDUs from a designated port that is not on the shortest-path tree. In other words, they are alternate looped paths to the root bridge that are longer than the path used by the root port. Blocking ports do not send BPDUs in traditional Spanning Tree Protocol; instead, a blocking port receives a constant flow of BPDUs from the neighboring designated port.

Task 1

Configure all switches to run 802.1D Spanning Tree Protocol.

Task 2

Ensure that SW1 is the root for every VLAN.

Task 3

Ensure that all switches wait a total of 10 seconds in the listening and learning states before moving a port to forwarding.

Task 4

Ensure that if the current root bridge fails, SW2 becomes the new root for all VLANs.

Task 5

Ensure that SW2's and SW3's E1/2 interface is used as the root port for all VLANs. Do not modify cost to achieve this.

Task 6

Ensure that SW5 uses its E3/1 port as the root port for VLANs 10 and 30. Do not modify SW2 to achieve this.

Task 7

Ensure that SW4 uses its E3/1 port as root port for VLANs 20 and 40. Do not modify SW3 to achieve this.

Task 8

Ensure that interfaces connected to non-switch hosts come online immediately.

a. If SW4 or SW5 detects a switch on these ports when they firstcome online, it should process the BPDUs normally. Otherwise, it should not process received BPDUs.
b. If SW6 detects a switch on one of these ports, it should disable theport.

Task 9

Ensure that SW6 is able to fully utilize redundant links between neighboring switches. Use a Cisco-specific approach to solve this.

Task 10

Ensure that SW6's interface toward SW5 is used as the root port for all VLANs. If the link between SW5 and SW6 goes down, SW6 should immediately switch to using its link toward SW4.

Task 11

Ensure that SW3's interfaces is designated on the SW2/SW3 link for all VLANs.

Task 12

Ensure that SW4 and SW5 can recover from an indirect link failure within 10 seconds. Do not modify spanning-tree timers.

Task 13

Ensure that SW2 only allows its ports toward SW1 to become root ports.

Note

Do not forget to reset the switch configurations to initial configurations upon completing this exercise.

802.1w Per-VLAN Rapid Spanning Tree Protocol

So far, we have explored the processes and functionality of traditional Spanning Tree Protocol (also known as 802.1D). The original intent of 802.1D was to ensure that loop-free paths exist in an L2-switched network where redundant links are utilized. 802.1D in its base form accomplishes this goal, but it does so at a price: convergence time. 802.1D utilizes timer-based convergence mechanics, which means ports must receive and evaluate BPDUs to determine the root bridge.

The winning BPDU is stored by each port and echoed out all designated ports on the switch until it is refreshed by the reception of the same BPDU on one

of the switch's non-designated (blocking or root) ports. If a port does not receive a BPDU within the Max Age time, the BPDU is aged out, and the topology reconverges. Furthermore, a port that transitions from blocking to forwarding must first pass through listening and learning states.

The entire convergence process for 802.1D, with default timers, can take up to 50 seconds (20 seconds for Max Age time and 30 seconds for transitioning between listening and learning states). As mentioned earlier in the lab, this is a considerable amount of downtime for a modern network, interfering with host operations such as acquiring a network address to use for data communication.

Recognizing this inefficiency, the IEEE developed the **802.1w** standard, which is also called the **Rapid Spanning Tree Protocol**, or **RSTP**. The goal of RSTP is to drastically reduce the amount of time it takes a network to converge during a convergence event and whenever a switch is newly joined to a network. To do so, RSTP makes a few changes to the 802.1D mechanics:

- BPDUs are sent by all switches independently of reception of the rootbridge's BPDU.
- Listening and learning states are combined into a single **learning state**.
- Port roles more clearly define what function a specific port plays in thenetwork.
- Timer-based convergence is replaced by a proposal/agreement process.

In 802.1D, switches do not originate BPDUs. Instead, they relay the received superior BPDU from their root port out their designated ports. This superior BPDU is first originated by the root bridge and acts as the heartbeat of the spanning tree. 802.1w modifies this behavior. All RSTP-compliant switches generate their best stored BPDUs out all of their designated ports at each hello interval, regardless of whether or not one was received on the root ports. This transforms the BPDU from being a measurement of the activity of the root bridge to being a keepalive between two bridges.

This modification of BPDU generation means RSTP can determine if a neighboring switch is active, based on when it last received a BPDU from the neighboring switch. If three Hello intervals of BPDUs are missed (2-second intervals, for 6 seconds total), the switch can immediately act. For blocking ports, that action is to become designated and send its own BPDU. Thus, in order for a blocking port to remain in the blocking state, it must continue to receive superior BPDUs from its upstream designated port.

In addition, the port states were revised. As mentioned earlier, there is little difference between a port that is blocking and a port that is listening or learning. In each of these states, one of three actions is being performed:

- The port is not forwarding traffic (that is, it is discarding traffic).
- The port is learning about the BPDU topology while not forwardingtraffic.
- The port is learning MAC addresses while not forwarding traffic.

The first two actions correspond to the port refusing to process data frames even to the point where MAC addresses are not being learned over the ports. Instead, state information about the spanning-tree topology is being evaluated. These two functions fall within the purview of the blocking and listening states of the original 802.1D. In 802.1w, they are combined into a simple discarding state to signify that data traffic is being discarded while spanning-tree BPDUs are still being processed. The last point corresponds to the switch processing MAC address information to build MAC address tables on the interfaces—a function of the learning state. This function was deemed necessary and has been retained as the learning state in RSTP.

With these modifications, RSTP possesses only three states: **discarding**, **learning**, and **forwarding**. These states describe what the port is actively doing but do not describe what function in the spanning-tree topology these ports serve. This distinction is necessary to allow rapid convergence in special circumstances. For this, RSTP utilizes unused fields in the 802.1D BPDU (the flags field) to carry the port state and new port roles describing both what action the port is taking and what role that port has in the spanning-tree topology.

The port roles are:

- **Root:** The port that receives the best BPDU of all BPDUs received by the local switch
- **Designated:** The port that sends the best BPDU on its LAN segment
- **Alternate:** The port that receives a superior BPDU on the LAN segment; it is a potential replacement for the root port
- **Backup:** The port that receives the superior BPDU of the local switch's own designated port; it is a potential replacement for the switch's own designated port

Of all of the states, root and designated are the same and correspond to the root and forwarding states in 802.1D. Alternate and backup ports are synonymous with blocking ports in 802.1D, but their roles more clearly define where the port lies in the spanning tree. An alternate port is a port that receives a superior BPDU from another bridge that is not the best BPDU the switch has heard. Such ports provide alternative paths to the current root bridge.

A backup port is a port that is self-looped back to the sending local switch. This port could be connected to the same L2 segment that does not speak Spanning Tree Protocol. For example, if a switch is connected to a set of hosts through a hub, the hub echoes all received frames out all ports except the port on which the frames were originally received. For redundancy, the switch could connect to the hub through two ports. If such a situation occurs, the BPDU sent by the switch on port A connected to the hub would be echoed and received on port B, connected to the same hub. If port A is determined to have the superior BPDU, port B would have the backup role because it provides a redundant path to a LAN segment that does not lead back to the root bridge.

A port in 802.1w can be in any mixture of states and roles. For instance, a port can be in the designated discarding role/state, which means it is a port that the switch believes should be designated but has not transitioned to forwarding. This fact is important for RSTP's convergence algorithm.

In 802.1D, convergence is based on a timer-driven state machine. When a switch comes online, it sends BPDUs claiming to be root. When it receives a superior BPDU, the ports must transition from blocking, to listening, to learning, and finally to forwarding, based on the Forward Delay timer (which defaults to 15 seconds, for 30 seconds total). If a switch were to lose connection to the root port, it would announce itself as root toward its neighbors. The neighbors would ignore this information for the Max Age period (which is 20 seconds by default) before reacting to the topology change. The goal is to allow the network to converge before a port is placed into the forwarding state.

802.1w does not use the same process as 802.1D but uses a newer proposal and agreement process that goes as such:

1. A new link port on a switch tries to move from the blocking state to theforwarding state.
2. The port receives a superior BPDU from the root bridge.
3. All other non-edge ports are blocked on the local switch.
4. Once all other switch ports are blocked, the local switch authorizes theroot switch to put its port into the forwarding state.
5. The same process occurs on all of the local switch's remaining nonedge ports.

In **step 1**, the new link initializes in the **designated discarding** state. It exchanges BPDUs with the current root's port (which is also in the designated discarding state). During this time, both switches send a BPDU with the proposal bit set as an indication that they want their ports to become the designated port on the segment. Upon receipt of the superior BPDU at step 2, the local switch knows where its root port lies.

In **step 3**, the switch must ensure no loops can occur in the network based on this new information. To do so, it places all of its non-edge ports into the discarding state; this is called synchronization. Throughout this process, the root switch's port is still in the designated discarding state.

At **step 4**, the local switch tells the root switch, through a BPDU with the agreement flag set, that the root switch's port can be moved to the

designated forwarding state. This happens because the local switch has blocked all of its other non-edge ports, preventing any bridging loops.

At **step 5**, the local switch sends a BPDU with the proposal bit set out all of its remaining designated discarding ports to start the rapid convergence process with other potential spanning-tree bridges downstream from the root. Step 5 repeats the proposal/agreement process with each switch to which the local switch has a direct connection. The same process occurs: The switches exchange proposal BPDUs, the losing switch enters the synchronization state, and it informs the designated switch it can move its port to the designated forwarding state. In this way, the synchronization process flows downstream from the root to the edge of the network.

This process relies heavily on the switch determining which ports are edge ports and which are non-edge ports. RSTP keeps track of this by assigning an edge variable and link type status to each switch port. The edge variable indicates whether or not the port leads to an end host or sits at the edge of the network. Such ports do not connect to other switches and should not receive BPDUs. If an edge port receives a BPDU, it loses its edge status. Edge ports are allowed to immediately transition to a forwarding state; this is similar to the spanning-tree PortFast feature.

Link type refers to whether or not the switch port can use rapid transitions, as previously indicated. There are two link types: point-to-point and shared. A point-to-point link is connected to exactly one other RSTP-compliant switch. A shared port is connected to a shared LAN segment that utilizes hubs or repeaters and cannot transition rapidly, as in the previous proposal agreement process.

Switches attempt to detect link type by using the duplex setting of the interface: Half-duplex is considered shared, and full-duplex is considered point-to-point.

802.1w also achieves rapid convergence by modifying what constitutes a topology change in the network. In 802.1D, loss of a port or a port transitioning to blocking is considered a topology change event. In 802.1w, only a non-edge (blocking or alternate) port transitioning to the forwarding state generates a topology change event. The reason is that the new non-

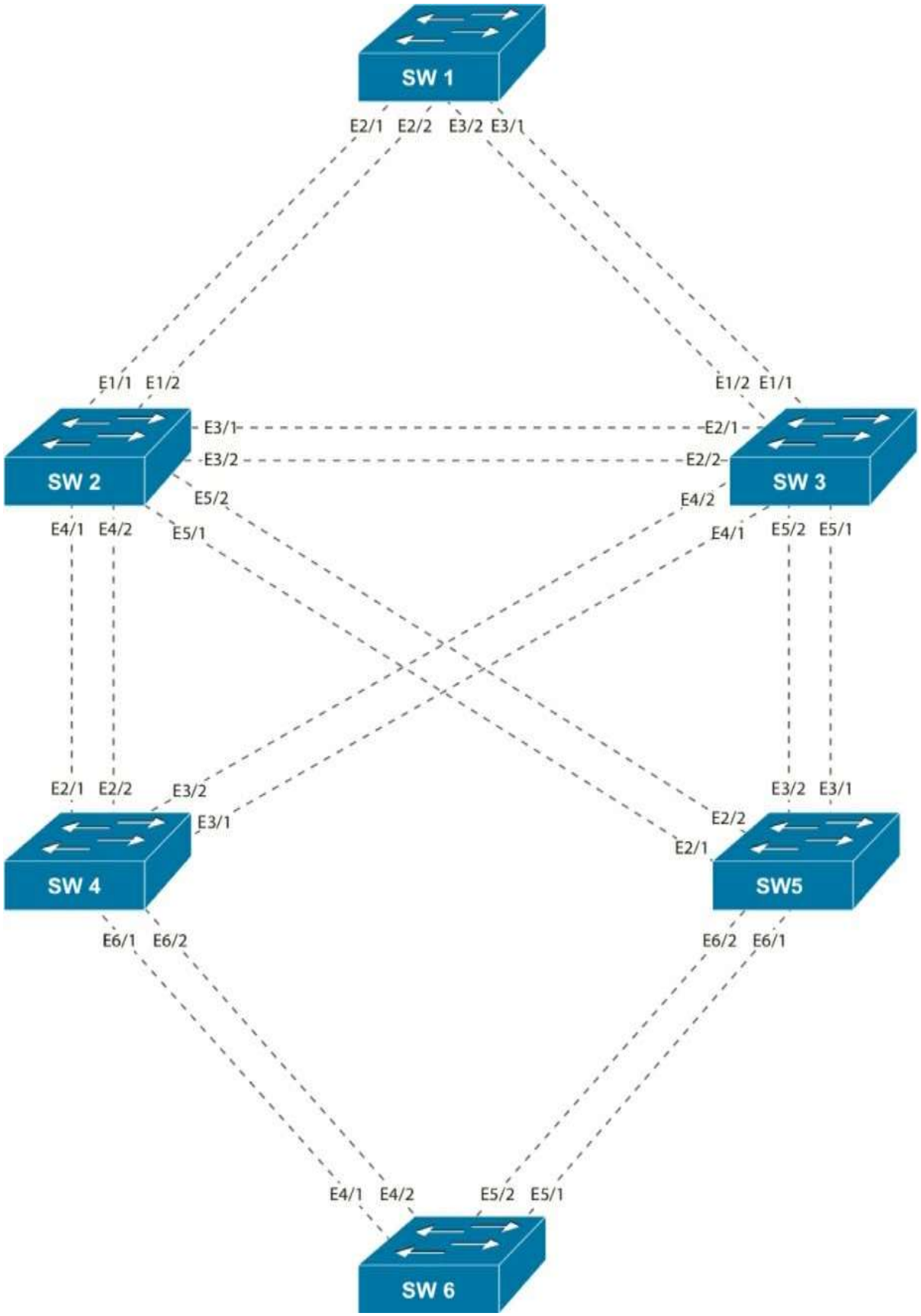
edge port offers a new path in the network, and the remaining switches should synchronize their MAC address tables accordingly to reflect the change. The TC-while timer is started on the switch initiating the topology change. During this time, the switch sends BPDUs with the TC flag set out all of its designated and root ports.

Switches that receive these BPDUs immediately age out all MAC addresses on all ports except where the TC BPDU was received. This mechanism allows rapid transition of the topology because TC BPDUs are originated by the switch experiencing the topology change event and are not initiated by the root bridge, as is the case in 802.1D.

Finally, because BPDUs are necessary for a blocking port to remain blocking, if a blocking port receives an inferior BPDU, it can react immediately to the information. This is in contrast to the 802.1D specification, which requires the stored BPDU on a port to age out before the switch converges the topology. In 802.1w, the switch receiving the inferior BPDU can infer that a topology change event has occurred somewhere in the network.

If a functioning root port exists on the switch receiving an inferior BPDU, it can simply respond with a proposal BPDU on its formerly blocking port, asking to be set to designated. This allows the failed switch to recover rapidly. If there is no functioning root port (meaning the inferior BPDU was received on a root port), the switch can assume that it should be the new root switch and indicates that with all of its remaining, now downstream, neighbors.

These are the key enhancements to 802.1D built into 802.1w. Some of them may sound familiar as they relate to many of the Cisco enhancement features to 802.1D, such as PortFast and Backbone Fast. The following lab demonstrates these enhancement features of 802.1w and contrasts them with their 802.1D equivalents.


SW 1
E2/1 E2/2 E3/2 E3/1
E1/1 E1/2
E1/2 E1/1
SW 2
E3/1
E3/2
E2/1
E2/2
SW 3
E5/2
E4/2
E4/1 E4/2
E5/1
E4/1
E5/2 E5/1
E2/1 E2/2
E3/2
E3/1
E3/2 E3/1
E2/2
E2/1
SW 4
SW5
E6/1 E6/2
E6/2 E6/1
E4/1 E4/2
E5/2 E5/1
SW 6

This lab should be conducted on the Mock Rack.

Initial Configuration:

Task 1

Change the hostname on each switch to SW#, where # is the number of the switch (for example, Switch 1 = SW1).

Task 2

Ensure that only the following ports on the switches are in an up/up state:

- SW1
- E2/1-2
- E3/1-2
- SW2
- E1/1-2
- E3/1-2
- E4/1-2
- E5/1-2
- SW3
- E1/1-2
- E2/1-2
- E4/1-2
- E5/1-2
- SW4
- E2/1-2
- E3/1-2

- E6/1-2
- SW5
- E2/1-2
- E3/1-2
- E6/1-2
- SW6
- E4/1-2
- E5/1-2

Task 3

Configure VLANs 10, 20, 30, and 40 on each switch, using any method.

Task 4

Configure trunk ports on all up/up interfaces.

802.1w Configuration Tasks

Task 1

Configure all switches to run 802.1w Rapid Spanning Tree Protocol.

Task 2

Ensure that SW1 is the root for all VLANs.

Task 3

Ensure that if the current root bridge fails, SW2 becomes the new root for all VLANs.

Task 4

Ensure that SW2's and SW3's E1/2 interface is used as the root port for all VLANs. Do not modify cost to achieve this.

Task 5

Ensure that SW5 uses the following ports as root:

- E2/1 for VLAN 10
- E2/2 for VLAN 20
- E3/1 for VLAN 30
- E3/2 for VLAN 40

Do not modify SW5 to achieve any of this.

Task 6

Ensure that SW4 uses its E3/1 port as the root port for VLAN 20.

Task 7

Ensure that RSTP features are enabled on all ports.

Task 8

Ensure that interfaces connected to non-switch hosts come online immediately.

a. If SW4 or SW5 detects a switch on these ports when it first comesonline, it should process the BPDUs normally. Otherwise, it should not process received BPDUs.

b. If SW6 detects a switch on one of these ports, it should disable the port.

Task 9

Ensure that SW6's links toward SW5 are used as the root port for all VLANs except VLAN 10. If the link between SW5 and SW6 goes down, SW6 should immediately switch to using its links toward SW4.

Task 10

Ensure that SW4 and SW5 can recover from an indirect link failure within 10 seconds. Do not modify the spanning-tree timers.

One of the major drawbacks of 802.1D spanning-tree was the timer-based convergence mechanisms. If a designated bridge loses its connection to the root, it will begin advertising itself as the root bridge to all of its connected ports. A downstream port in the blocking state would receive these inferior BPDUs. In 802.1D, the blocking port would completely ignore the information until the stored BPDU from the real root on that blocking port expires (in 20 seconds, based on the default Max Age timer). Then the switch signals a topology change event, and the port transitions to the forwarding state. The entire process can take up to 50 seconds with default timers.

Cisco's Backbone Fast feature was created to help this process by utilizing the RLQ protocol when a blocking port suddenly starts receiving inferior BPDUs. The RLQ messages are sent out in order to determine if a path to the root exists on the switch. If a path is found, the previously blocked port can bypass the Max Age timer and begin the transition to the forwarding state, cutting 20 seconds of convergence time.

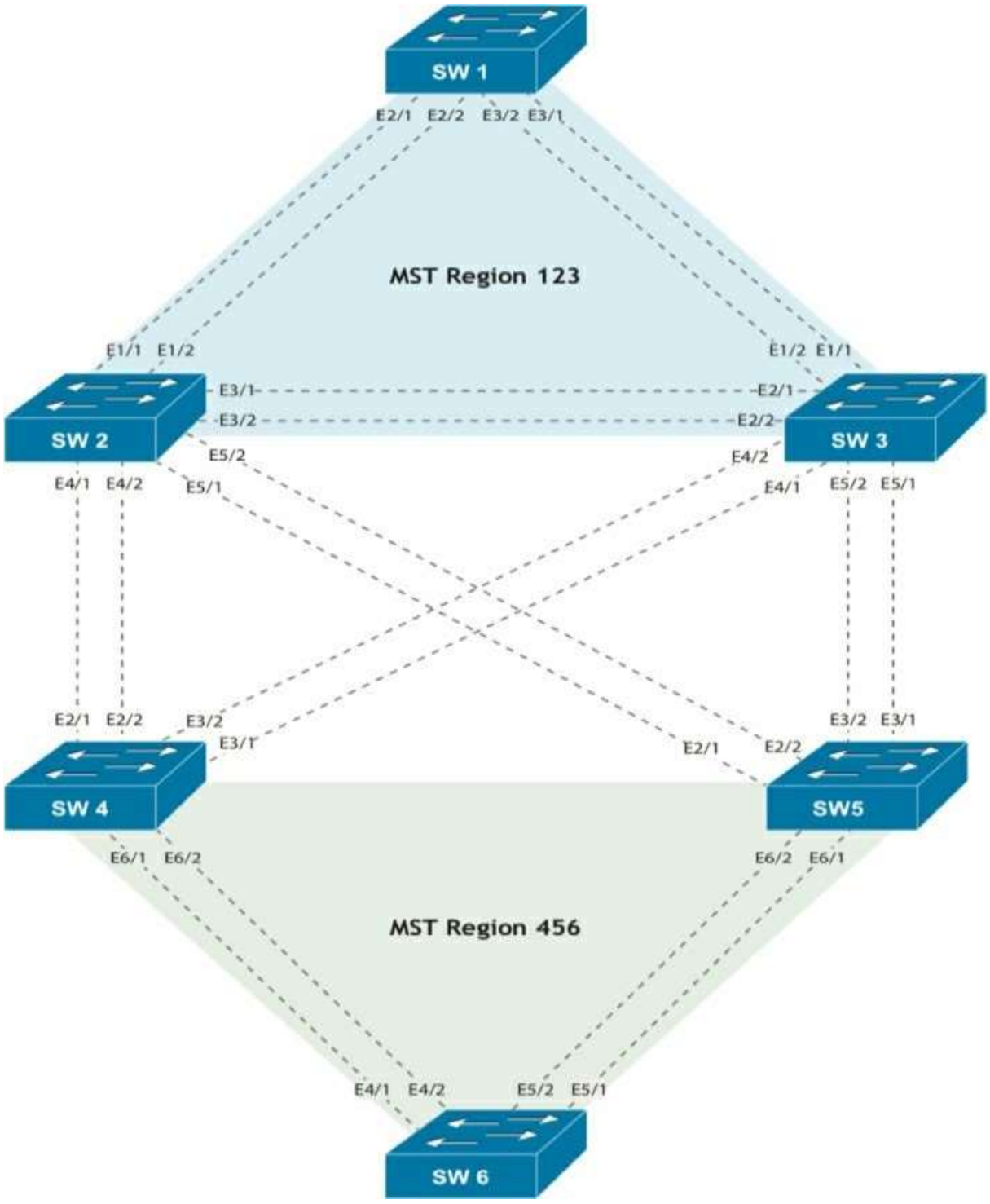
RSTP incorporates this function through its behavior of immediately accepting inferior BPDUs that are received on a discarding port. An inferior BPDU being received on a discarding port signifies that a topology change has occurred somewhere in the network. Either the current root has failed and the local switch has stale information or the upstream designated bridge has lost its connection to the root. In either case, the spanning-tree topology needs to reconverge and can do so rapidly by using the proposal/agreement processes.

Since this feature is built into RSTP, there is no configuration needed to accomplish this task.

Note

Do not forget to reset the switch configurations to initial configurations upon completing this exercise.

802.1s Multiple Spanning Tree Protocol



This lab should be conducted on the Mock Rack.

Initial Configuration:

Task 1

Change the hostname on each switch to SW#, where # is the number of the switch (for example, Switch 1 = SW1)

Task 2

Ensure that only the following ports on the switches are in an up/up state:

- SW1
- E2/1-2
- E3/1-2
- SW2
- E1/1-2
- E3/1-2
- E4/1-2
- E5/1-2
- SW3
- E1/1-2
- E2/1-2
- E4/1-2
- E5/1-2
- SW4
- E2/1-2
- E3/1-2
- E6/1-2
- SW5
- E2/1-2
- E3/1-2

- E6/1-2

- SW6
- E4/1-2
- E5/1-2

Task 3

Configure VLANs 10, 20, 30, and 40 on each switch, using any method.

Task 4

Configure trunk ports on all up/up interfaces.

Let's Explore 802.1s

One of the major drawbacks to spanning tree, be it the original Spanning Tree Protocol or RSTP, is the fact that redundant links are blocked in the network, making their bandwidth useless unless a failover event occurs. Cisco alleviated this impact by implementing PVST/+ and PVRST/+, which allow administrators to load balance individual VLANs on trunk links to better utilize the bandwidth.

The issue with this approach is that the switch must calculate a separate spanning tree for each VLAN independently. As the total number of VLANs grows, the switch spends more processor cycles calculating the spanning-tree topology. This is inefficient as the total number of possible spanning-tree topologies is limited by the total number of switches and links in the network. If there are only three switches in the network, then there are effectively only three total topologies that can exist: one topology with each switch as the root of the spanning tree. If there are a total of four VLANs in the same sample network, it is reasonable to say that the fourth VLAN spanning-tree calculation yields a redundant result to one of the previous three.

The IEEE 802.1s Multiple Spanning Tree Protocol (MST) standard mitigates this by providing a standards-based way to run different instances of Spanning Tree Protocol on a single switch. Instead of running a single instance for each VLAN, the administrator can define the instances of Spanning Tree Protocol that will be run and their parameters, such as priority and costs. Multiple VLANs are mapped to the configured instances, allowing one instance to represent one or many VLANs.

With this construct, an administrator can group together VLANs with similar pathing requirements in a single instance rather than tuning individual VLAN-specific spanning-tree instance settings. It is even possible to map all VLANs to a single instance and reduce spanning-tree processing overhead for all VLANs. Doing this reduces the computational load on the switches in the network.

The most noticeable way a switch saves computational cycles is through MST's use of a single BPDU for all instances. Instead of explicitly sending a BPDU for each VLAN or each instance containing VLAN-to-instance mapping information, MST exchanges a single BPDU. This BPDU contains information about the root bridge and special M-records. These M-records carry the spanning-tree topology information for all instances configured on the switch.

A switch running MST must be explicitly configured with the VLAN-toinstance mapping in its MST configuration. The MST configuration first gives the MST process a name. This process name is similar to the EIGRP autonomous system number. Then, the process is explicitly given a revision number. Finally, the VLAN-to-instance mapping is configured.

Note

Unlike with EIGRP, there can be only one MST process running multiple instances at a time on a switch. Cisco documentation calls the MST process an *MST instance*, but for clarity between MST instances that have VLANs mapped to them, the term *process* is used in this guide.

In order to operate, an MST network needs to be configured with the instances that exist in the network and what VLANs are mapped to those instances. If, for example, instance 1 on SWA contains VLAN 1 but instance 10 on SWB contains VLAN 1, SWA and SWB could come to different conclusions about whether or not the port is forwarding or blocking for both VLANs. However, because no explicit VLAN-to-instance mapping
information is carried between switches, there is no way for

MST validates VLAN-to-instance mapping consistency between two bridges by including the MST process, the revision number, and a digest of the MST VLAN-to-instance mapping table in the BPDU. If the received BPDU matches a switch's internal configuration, the switch knows it can trust the information contained in the M-records of the BPDU. The set of switches in a network that all have matching MST process names, revision numbers, and VLAN-to-instance mapping tables is called an *MST region*. Multiple MST regions can exist in a network. Topology changes in one region do not affect other regions. Also, each region builds an internal spanning tree separately as well as cooperatively with other regions to form one loop-free path throughout the entire switched network.

Within a region, there always exists MST instance 0, which is called IST0. This instance is the only instance that interacts with other regions and is the instance to which all VLANs are initially mapped when MST is first enabled on a switch. IST0 elects a root bridge called the *IST0 root*. A non-root switch calculates a loop-free path from itself to the IST0 root. This ensures that no matter how many instances are added or VLANs are moved around, there is always at least one loop-free path in the MST topology.

Switches exchange IST0 BPDUs that contain the M-records for all other instances in the MST region. Using these records, independent spanning-tree topologies can be calculated for each instance.

IST0 and M-records help create a loop-free path within an MST region. MST uses a separate process to help negotiate a loop-free path between different MST regions. Switches that connect to neighboring switches in different regions are called *MST boundary switches*. The specific port that the local switch shares with the remote neighboring switch in a different region is called the *boundary port*. On boundary ports, MST only exchanges IST0 BPDU

information. This makes sense because IST0 always exists within the MST region and is guaranteed to be loop free at all times. The receiving switch compares the IST0 bridge information to determine if it is superior or inferior to its own IST0 BPDU. This spanning tree running between regions is called the common spanning tree, or CST.

It helps at this point to consider two MST regions as single switches. Because the boundary switches hide the internal M-record topology from switches in different MST regions, the topology becomes very simple from the receiving switch's perspective. Because the receiving switch cannot ensure that VLANs are mapped to the same instances as its own, the receiving switch must make a blocking/forwarding decision on its boundary port for all VLANs. This is the only way to ensure a loop-free path to the remote MST region. The MST boundary switches only compare the IST0 information received from other MST boundary switches. The boundary switches analyze this information and make forwarding or discarding decisions for all VLANs on the boundary ports.

MST interacts with regular 802.1D spanning-tree regions in a similar way. A boundary switch compares a received BPDU with its own IST0 BPDU to make a forwarding or discarding decision for all VLANs on the boundary port. This process ensures that redundant inter-region links are blocked between regions and not internally within a specific region—in the same way a switch blocks ports between switches and not links within the switch itself.

MST interacts with PVST regions a bit differently. This interaction is explained further later on in this solution guide.

Task 1

Configure all the switches to utilize the following spanning-tree configuration for SW1, SW2, and SW3:

- MST is the operational mode.
- The region name is 123.
- The revision number is 1.
- Configure the following instance-to-VLAN mappings:

- Instance 1: VLANs 10–19
- Instance 2: VLANs 20–29
- Instance 3: VLANs 30–39
- Instance 4: VLANs 40–49

Task 2

Configure all the switches to utilize the following spanning-tree configuration for SW4, SW5, and SW6:

- MST is the operational mode.
- The region name is 456.
- The revision number is 1.
- Configure the following instance-to-VLAN mappings:
- Instance 1: VLAN 10
- Instance 2: VLAN 20
- Instance 3: VLAN 30
- Instance 4: VLAN 40

Task 3

Ensure that SW1 is the root for the entire topology.

Task 4

Configure Region 123 as follows:

- SW1 should be the IST root for Instances 1 and 4.
- SW2 should be the IST root for Instance 2.
- SW3 should be the IST root for Instance 3.

Configure Region 456 as follows:

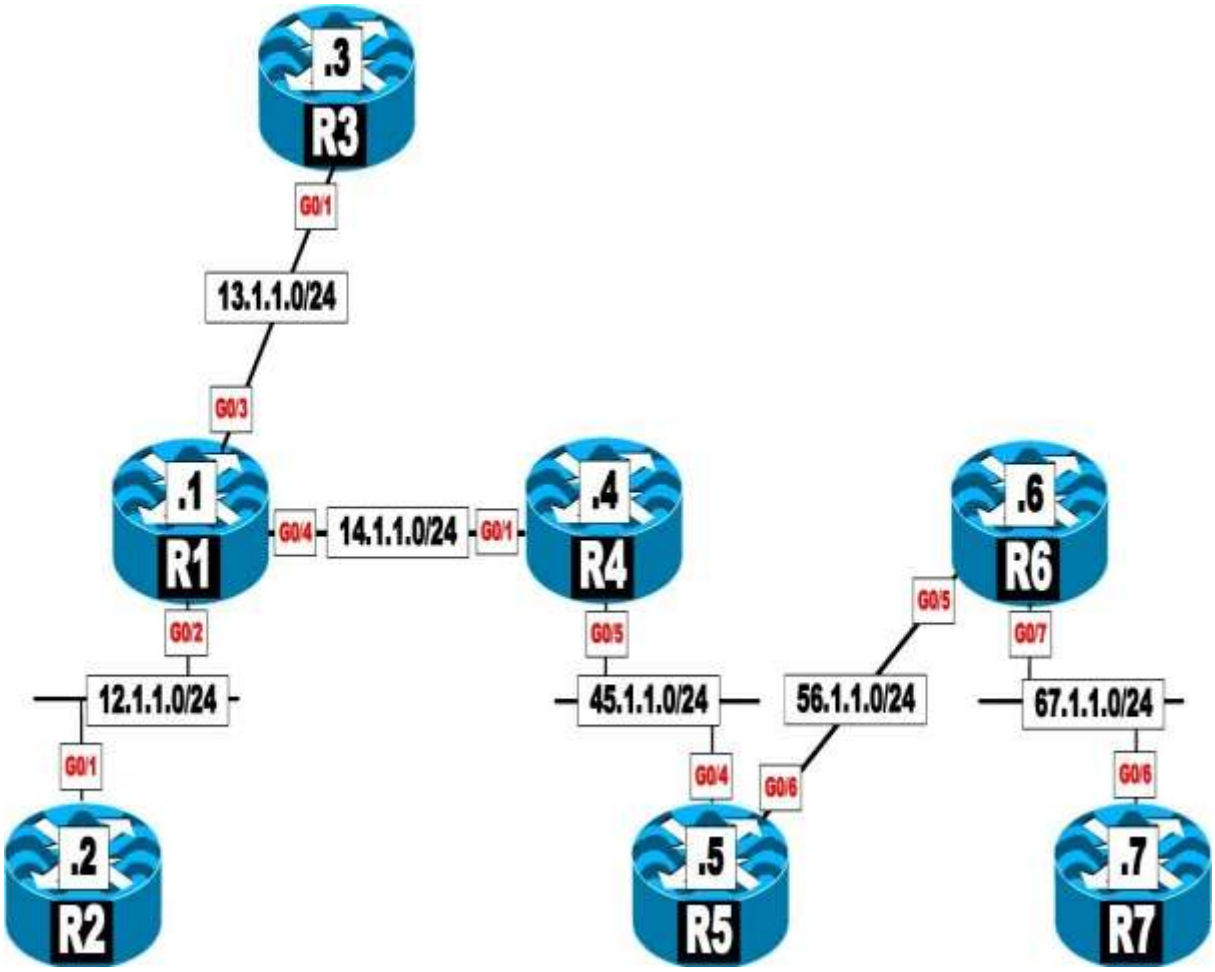
- SW6 should be the IST root for all instances except Instance 0.
- SW5 should be the regional root.

Chapter 2. DMVPN [This content is currently in development.]

This content is currently in development.

Chapter 3. IP Prefix Lists

Lab 1: Prefix Lists



This lab should be conducted on the enterprise rack.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use " Lab 1-Prefix-Lists" from the "IP Prefix-lists" folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → IP-Prefix-list folder → Lab-1.

Task 1

Configure R1 to filter 192.1.1.32/27 (an OSPF PID 1 route) using a prefix list through its G0/2 interface.

Task 2

Configure R1 such that it only permits class A networks that are not subnetted and filter the rest of the prefixes/networks. These are RIPv2 routes advertised by R3.

Task 3

Configure R4 such that it only allows class B networks that are not subnetted. These are OSPF routes advertised by R1.

Task 4

Configure R5 such that it only allows class C networks that are not subnetted. These are EIGRP routes that are advertised by R4.

Task 5

Configure R4 such that it denies networks 10.4.4.33/27 and 10.4.5.65/26 and allows the rest of the networks. You should configure a minimum number of lines in the prefix list to accomplish this task. These are the RIPv2 routes advertised by R5.

Task 6

Configure R5 to inject a default route in the RIP routing domain. If this configuration is successful, R4 should see the default route in its routing table.

Task 7

R4 should be configured to filter the default route injected in the previous task.

Task 8

Configure R6 to filter any networks with a prefix length of 26 or less. These are OSPF routes advertised by R5.

Task 9

Reconfigure R6 to filter any networks with a prefix length of 26 or greater. These are OSPF routes received from R5.

Task 10

Configure R7 to filter the following networks:

- 146.1.2.129/25
- 146.1.3.193/26
- 146.1.4.225/27
- 6.1.4.225/27
- 6.1.5.241/28

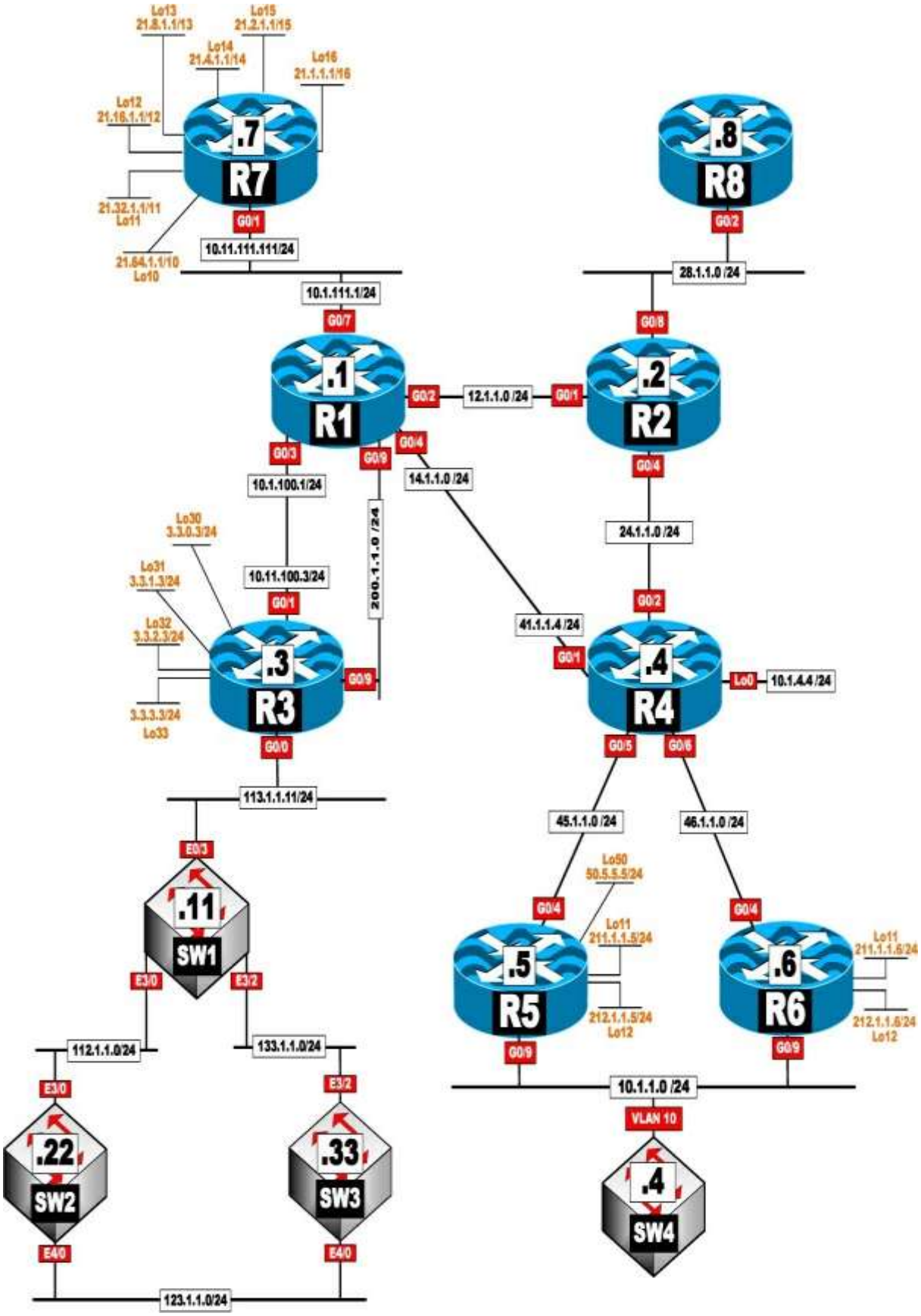
You should configure only three prefix list statements. These are EIGRP routes advertised by R6.

Task 11

Erase the startup configuration and reload the routers before proceeding to the next lab.

Chapter 4. RIPv2

Lab 1: Configuring RIPv2


Lo13
21.8.1.1/13
Lo15
21.2.1.1/15
Lo14
21.4.1.1/14
Lo16
21.1.1.1/16
Lo12
21.16.1.1/12
.7
R7
21.32.1.1/11
Lo11
21.64.1.1/10
Lo10
G0/1
10.11.111.111/24
.8
R8
G0/2
28.1.1.0 /24
10.1.111.1/24
G0/7
G0/8
.1
R1
G0/2
12.1.1.0 /24
G0/1
.2
R2
G0/3
G0/9
G0/4
14.1.1.0 /24
10.1.100.1/24
200.1.1.0 /24
G0/4
24.1.1.0 /24
Lo30
3.3.0.3/24
Lo31
3.3.1.3/24
10.11.100.3/24
G0/1
G0/2
41.1.1.4 /24
Lo32
3.3.2.3/24
.3
R3
G0/9
.4
R4
Lo0
10.1.4.4 /24
3.3.3.3/24
Lo33
G0/0
G0/5
G0/6
113.1.1.11/24
45.1.1.0 /24
46.1.1.0 /24
E0/3
Lo50
50.5.5.5/24
.11
SW1
G0/4
G0/4
Lo11
211.1.1.5/24
.5
R5
.6
R6
Lo11
211.1.1.6/24
E3/0
E3/2
212.1.1.5/24
Lo12
212.1.1.6/24
Lo12
G0/9
G0/9
112.1.1.0/24
133.1.1.0/24
10.1.1.0 /24
E3/0
E3/2
VLAN 10
.22
SW2
.33
SW3
.4
SW4
E4/0
E4/0
123.1.1.0/24

This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 1-Configuring RIPv2 in the RIPv2 folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-1.

Task 1

Configure RIPv2 on all routers and switches. Advertise their directly connected interfaces in this routing domain. These devices *should not* have a classful nature. If this configuration is successful, the routers and switches should successfully exchange routes.

Use the following policy when configuring this task:

1. R7 and R1 *must* use two static routes for reachability.
2. R1 and R4 *must not* use a static route(s). R1 *cannot* use any solution that's used in the previous policy or the next ones to provide reachability. You are allowed to disable the sanity check.
3. You *cannot* use PBR to resolve any of the policies in this task.
4. The IP addresses on the topology are correct.

Task 2

1. Configure R4 such that it sends RIPv2 updates out of its G0/3 interfaceto a broadcast destination.
    a. Do not change RIP's version.

2. Ensure that R1 is configured to send unicast updates to its neighboringrouter R7.

Task 3

Configure the following loopback interfaces on R3. Starting with network 30.3.0.0 /24, this router should *only* advertise every other third octet network.

(for example, x.x.0.x, x.x.2.x, x.x.4.x). These are networks 30.3.0.0, 30.3.2.0, 30.3.4.0, 30.3.6.0, 30.3.8.0, and 30.3.10.0/24.

int lo0 = 30.3.0.1 /24	int lo6 = 30.3.6.1 /24
int lo1 = 30.3.1.1 /24	int lo7 = 30.3.7.1 /24
int lo2 = 30.3.2.1 /24	int lo8 = 30.3.8.1 /24
int lo3 = 30.3.3.1 /24	int lo9 = 30.3.9.1 /24
int lo4 = 30.3.4.1 /24	int lo10 = 30.3.10.1 /24
int lo5 = 30.3.5.1 /24	

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-1-Task3.

Task 4

Configure the following loopback interfaces on R6. Starting with network 60.6.0.0 /24, this router should *only* advertise every eighth third octet subnet.

(for example x.x.0.x, x.x.8.x, x.x.16.x)

int lo0 = 60.6.0.1 /24	int lo6 = 60.6.6.1 /24
int lo1 = 60.6.1.1 /24	int lo7 = 60.6.7.1 /24
int lo2 = 60.6.2.1 /24	int lo8 = 60.6.8.1 /24
int lo3 = 60.6.3.1 /24	int lo9 = 60.6.9.1 /24
int lo4 = 60.6.4.1 /24	int lo10 = 60.6.10.1 /24
int lo5 = 60.6.5.1 /24	

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-1-Task4.

Task 5

Configure the following loopback interfaces on R4. This router should be configured such that it *only* advertises the even-numbered hosts of the odd third octet networks of these loopback interfaces plus all other networks. (for example x.x.1.x, x.x.3.x, x.x.5.x)

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-1-Task5.

int lo1	int lo6
ip addr 40.4.1.1 255.255.255.0	ip addr 40.4.6.1 255.255.255.0
ip addr 40.4.1.2 255.255.255.255 sec	ip addr 40.4.6.2 255.255.255.255 sec
ip addr 40.4.1.3 255.255.255.255 sec	ip addr 40.4.6.3 255.255.255.255 sec
ip addr 40.4.1.4 255.255.255.255 sec	ip addr 40.4.6.4 255.255.255.255 sec
ip addr 40.4.1.5 255.255.255.255 sec	ip addr 40.4.6.5 255.255.255.255 sec

ip addr 40.4.1.6 255.255.255.255 sec ip addr 40.4.1.7 255.255.255.255 sec ip addr 40.4.1.8 255.255.255.255 sec ip addr 40.4.1.9 255.255.255.255 sec ip addr 40.4.1.10 255.255.255.255 sec	ip addr 40.4.6.6 255.255.255.255 sec ip addr 40.4.6.7 255.255.255.255 sec ip addr 40.4.6.8 255.255.255.255 sec ip addr 40.4.6.9 255.255.255.255 sec ip addr 40.4.6.10 255.255.255.255 sec
int lo2 ip addr 40.4.2.1 255.255.255.0 ip addr 40.4.2.2 255.255.255.255 sec ip addr 40.4.2.3 255.255.255.255 sec ip addr 40.4.2.4 255.255.255.255 sec ip addr 40.4.2.5 255.255.255.255 sec ip addr 40.4.2.6 255.255.255.255 sec ip addr 40.4.2.7 255.255.255.255 sec ip addr 40.4.2.8 255.255.255.255 sec ip addr 40.4.2.9 255.255.255.255 sec ip addr 40.4.2.10 255.255.255.255 sec	**int lo7** ip addr 40.4.7.1 255.255.255.0 ip addr 40.4.7.2 255.255.255.255 sec ip addr 40.4.7.3 255.255.255.255 sec ip addr 40.4.7.4 255.255.255.255 sec ip addr 40.4.7.5 255.255.255.255 sec ip addr 40.4.7.6 255.255.255.255 sec ip addr 40.4.7.7 255.255.255.255 sec ip addr 40.4.7.8 255.255.255.255 sec ip addr 40.4.7.9 255.255.255.255 sec ip addr 40.4.7.10 255.255.255.255 sec
int lo3 ip addr 40.4.3.1 255.255.255.0 ip addr 40.4.3.2 255.255.255.255 sec ip addr 40.4.3.3 255.255.255.255 sec ip addr 40.4.3.4 255.255.255.255 sec ip addr 40.4.3.5 255.255.255.255 sec ip addr 40.4.3.6 255.255.255.255 sec ip addr 40.4.3.7 255.255.255.255 sec ip addr 40.4.3.8 255.255.255.255 sec ip addr 40.4.3.9 255.255.255.255 sec ip addr 40.4.3.10 255.255.255.255 sec	**int lo8** ip addr 40.4.8.1 255.255.255.0 ip addr 40.4.8.2 255.255.255.255 sec ip addr 40.4.8.3 255.255.255.255 sec ip addr 40.4.8.4 255.255.255.255 sec ip addr 40.4.8.5 255.255.255.255 sec ip addr 40.4.8.6 255.255.255.255 sec ip addr 40.4.8.7 255.255.255.255 sec ip addr 40.4.8.8 255.255.255.255 sec ip addr 40.4.8.9 255.255.255.255 sec ip addr 40.4.8.10 255.255.255.255 sec
int lo4 ip addr 40.4.4.1 255.255.255.0 ip addr 40.4.4.2 255.255.255.255 sec ip addr 40.4.4.3 255.255.255.255 sec ip addr 40.4.4.4 255.255.255.255 sec ip addr 40.4.4.5 255.255.255.255 sec ip addr 40.4.4.6 255.255.255.255 sec ip addr 40.4.4.7 255.255.255.255 sec ip addr 40.4.4.8 255.255.255.255 sec ip addr 40.4.4.9 255.255.255.255 sec ip addr 40.4.4.10 255.255.255.255 sec	**int lo9** ip addr 40.4.9.1 255.255.255.0 ip addr 40.4.9.2 255.255.255.255 sec ip addr 40.4.9.3 255.255.255.255 sec ip addr 40.4.9.4 255.255.255.255 sec ip addr 40.4.9.5 255.255.255.255 sec ip addr 40.4.9.6 255.255.255.255 sec ip addr 40.4.9.7 255.255.255.255 sec ip addr 40.4.9.8 255.255.255.255 sec ip addr 40.4.9.9 255.255.255.255 sec ip addr 40.4.9.10 255.255.255.255 sec
int lo5 ip addr 40.4.5.1 255.255.255.0	**int lo10** ip addr 40.4.10.1 255.255.255.0

ip addr 40.4.5.2 255.255.255.255 sec	ip addr 40.4.10.2 255.255.255.255 sec
ip addr 40.4.5.3 255.255.255.255 sec	ip addr 40.4.10.3 255.255.255.255 sec
ip addr 40.4.5.4 255.255.255.255 sec	ip addr 40.4.10.4 255.255.255.255 sec
ip addr 40.4.5.5 255.255.255.255 sec	ip addr 40.4.10.5 255.255.255.255 sec
ip addr 40.4.5.6 255.255.255.255 sec	ip addr 40.4.10.6 255.255.255.255 sec
ip addr 40.4.5.7 255.255.255.255 sec	ip addr 40.4.10.7 255.255.255.255 sec
ip addr 40.4.5.8 255.255.255.255 sec	ip addr 40.4.10.8 255.255.255.255 sec
ip addr 40.4.5.9 255.255.255.255 sec	ip addr 40.4.10.9 255.255.255.255 sec
ip addr 40.4.5.10 255.255.255.255 sec	ip addr 40.4.10.10 255.255.255.255 sec

Task 6

Configure the routers and switches in this routing domain based on the following timers:

- Periodic updates are sent every 30 seconds.
- Routers and switches should declare a route invalid after 1.5 minutes.
- Routers and switches should suppress routing information regarding abetter path for 1.5 minutes.
- Routers and switches should flush routes 30 seconds after they are invalid.
- Routers and switches should postpone their periodic updates by 100milliseconds.

Task 7

Since R1 is a very fast router, configure it such that it adds an inter-packet delay of 50 milliseconds between the updates.

Task 8

Configure R2 to set the number of received but *not* yet processed RIP update packets in the RIP input queue to 100.

Task 9

Configure all routers to suppress a flash update when a topology change occurs 10 seconds before a regular update.

Task 10

Configure R1 and R2 to authenticate their routing updates through their direct connection. Configure these two routers to use the unencrypted key **ccie** for this purpose.

Task 11

Configure R5, R6, and SW4 to use authentication with the strongest authentication method available to RIPv2. These routers should use **Micronic?** as their password.

Task 12

Configure R1 to accept existing and future routes that have a prefix length of 10 to 14. These routes should be received from R7 *only*. *Do not* use an access list(s) or a **neighbor** command to accomplish this task.

Task 13

Configure R7 to inject a default route.

Task 14

Configure R4 to filter the default route injected by R7.

Task 15

Configure SW4 such that it always prefers to reach network 10.1.4.0 /24 through R6.

1. SW4 should use R5 when R6 is down.
2. Restrictions: *Do not* use an offset list to accomplish this task.

Task 16

Configure SW4 to filter network 50.5.5.0 /24. *Do not* use an access list to accomplish this task.

Task 17

Configure R4 such that it injects a default route into the RIPv2 routing domain.

1. Restrictions:
    a. This default route should *not* be given to R6.
    b. *Do not* configure R6 to accomplish this task. R5 should *only* have a default route from R4.

Task 18

Configure R3 to summarize its Loopback Lo0–Lo3, Lo30–Lo33 and advertise two summary routes into the RIP routing domain.

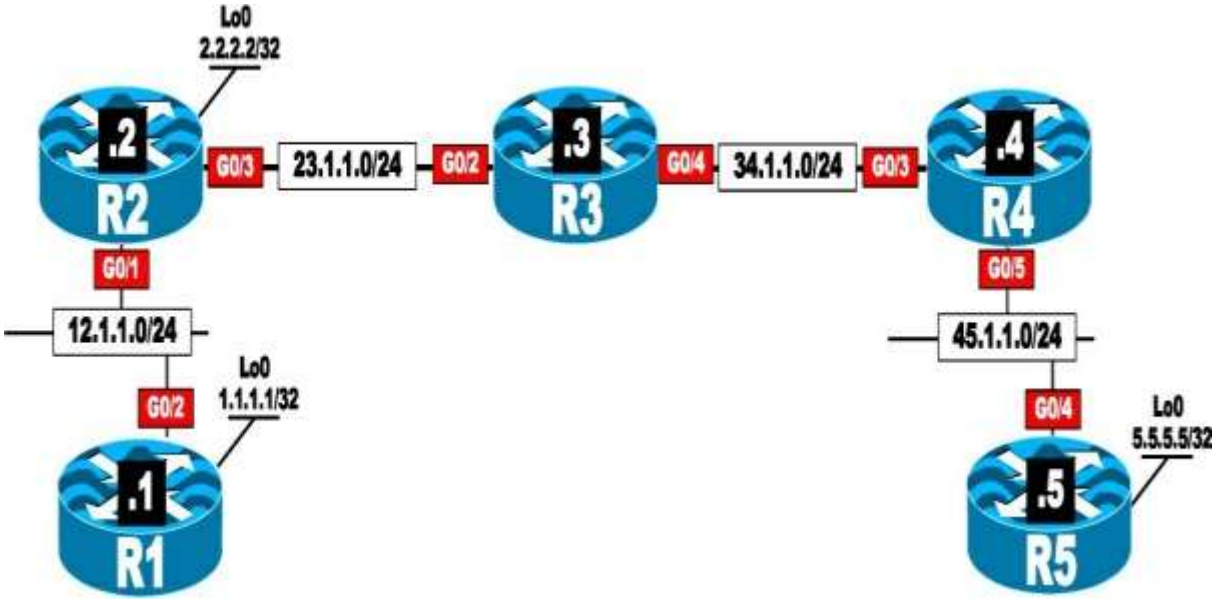
Task 19

Configure Lo200 with the IP address 120.2.2.2 /24 on SW2. This switch should advertise this network in the RIPv2 routing domain. Configure SW1 such that this network is never advertised to any router downstream/beyond SW4, as those are future devices connected to SW4.

Task 20

Erase the startup configuration of the routers, the startup configuration, and the VLAN.dat file for each switch and reload the devices before proceeding to the next lab.

Lab 2: Helper Map



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 2-Helper-map in the RIPv2 folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-2.

Task 1

Configure OSPF Area 0 on the following interfaces:

- The G0/1, G0/3, and loopback0 interfaces of R2
- All directly connected interfaces of R3
- The G0/3 interface of R4

Task 2

Configure RIPv2 on the:

- Lo0 and G0/2 interfaces of R1
- G0/0 interface on R5

Disable auto-summarization on these devices.

Task 3

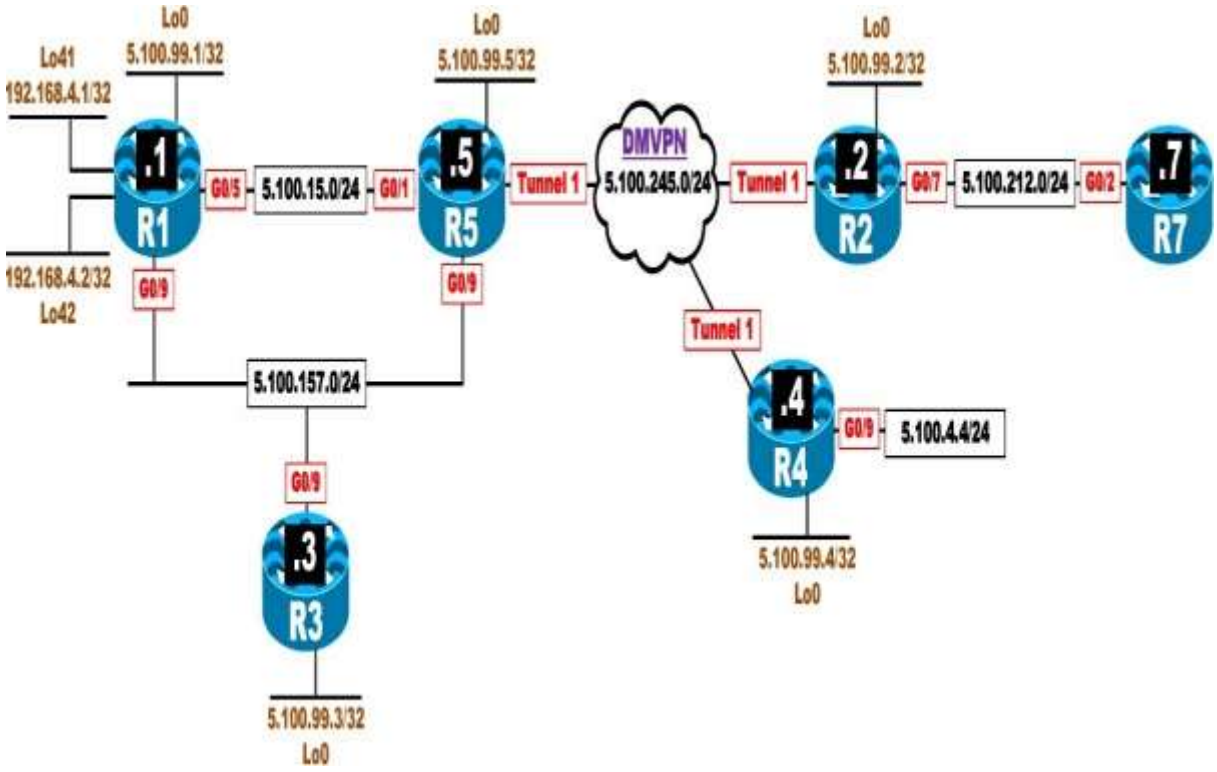
Configure multicasting on the appropriate routers such that R5 receives all the RIPv2 updates from R1.

- R2 should be configured as the RP and the BSR router. This routershould use its loopback interface as the source of all its BSR messages.
- You must use 224.1.1.1 to accomplish this task.
- Restrictions:
    a. *Do not* run multiple unicast routing protocols on any of the routers.
    b. *Do not* configure GRE, IPnIP, MPLS, LDP, or any type of tunneling to accomplish this task.

Task 4

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 3: RIPv2 Challenge Lab



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tickets and use Lab 3-RIPv2 Challenge Lab in the RIPv2 folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → RIPv2 folder → Lab-3.

Ticket 1

R1 is configured to filter its Lo41. However, this interface is still reachable from R5.

- Restrictions:
  a. *Do not* configure another access list, prefix list, or route map.
  b. Use only two commands to accomplish this task.

Ticket 2

R7 is configured to filter all even third octet networks (for example x.x.2.x, x.x.4.x, x.x.6.x). with the mask /24. However, this has affected all routes, and none of them are reachable from R2.

Ticket 3

R4 can't reach R1's Lo42 using its Lo0 as the source:

- Use a single command to fix this problem.

Ticket 4

R2 is configured to filter its G0/7 interface with the IP address 5.100.212.2/24, but R5 can't reach R2's Lo0.

Ticket 5

R4 can't reach R2's Lo0 interface.

- Restrictions:
  a. *Do not* configure the **neighbor** command.

  b. *Do not* change the DMVPN phase or configure DMVPN in a

dynamic manner.

Ticket 6

R3 can't ping R1's Lo0.

Ticket 7

R3 is configured to filter all routes received from R5. However, the routes are still there. *Do not* use another method to fix this problem; correct the existing problem.

Ticket 8

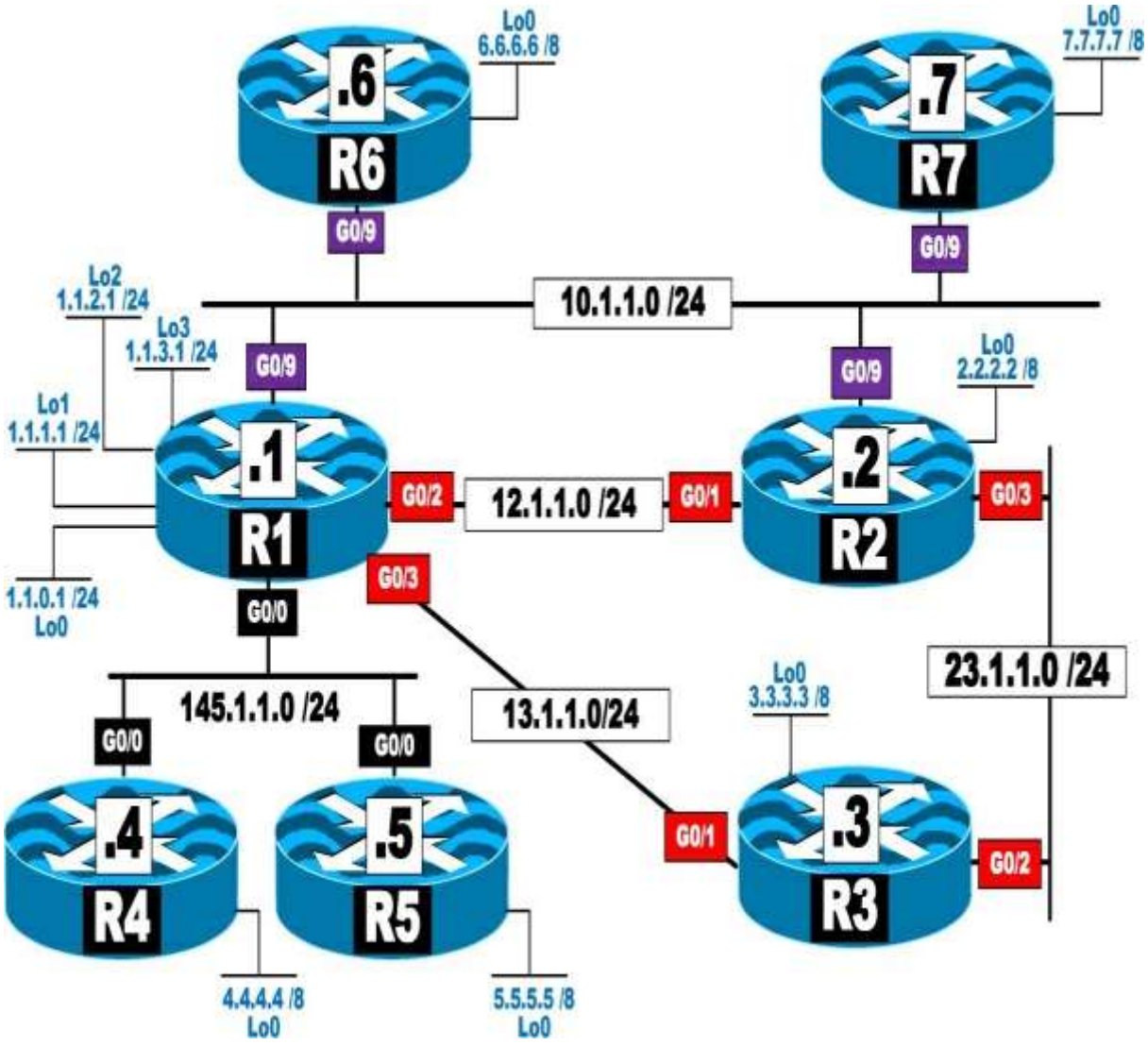
R7 should *not* have any RIPv2 routes in its routing table. You *must* configure an outbound filter using a standard numbered ACL and a **distribute-list** command on the G0/7 interface of R2 to accomplish this task. You are allowed to remove one command.

Ticket 9

Erase the startup configuration and reload the devices before proceeding to the next lab.

Chapter 5. EIGRP

Lab 1: EIGRP Named Mode



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 1-EIGRP Named Mode in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-1.

Task 1

Configure EIGRP on R1, R2, and R3 based on the following policy:

Router	Interface	AS Number
R1	G0/9 G0/0 G0/2 G0/3 Loopback0– Loopback3	200 100 100 100 100
R2	G0/9 G0/1 G0/3 Loopback0	200 100 100 100
R3	G0/1 G0/2 Loopback0	100 100 100

- R1 should be configured to use unicast to establish a neighbor adjacency with R2.
- R1 should use multicast to establish a neighbor adjacency with R3.
- R1, R2, and R3 should use an EIGRP named mode configuration toaccomplish this task.

Task 2

Configure R4 and R5 in EIGRP AS 100. You must use named mode to accomplish this task.

Task 3

Configure R1, R4, and R5 to use unicast to establish neighbor adjacency.

Task 4

Configure R6 in EIGRP AS 200. This router should run EIGRP AS 200 on its G0/9 and Loopback0 interfaces. You should use an EIGRP named mode configuration to accomplish this task.

Task 5

Configure OSPF Area 0 on R6's G0/9 and R7's G0/9 and Loopback0 interfaces. The router ID of these routers should be configured as 0.0.0.*x*, where *x* is the router number.

Task 6

Configure R6 to redistribute OSPF into EIGRP such that R1 and R2 go directly to R7 to reach the 7.0.0.0/8 network.

Task 7

Configure the hello interval of all routers in AS 200 to be twice the default.

Task 8

Configure R4 such that in the worst-case scenario, it uses 10% of the bandwidth for its EIGRP updates. This policy should apply to the existing and future interfaces.

Task 9

Configure R1 to summarize its loopback interfaces and advertise a single summary in the EIGRP AS 100 routing domain.

Task 10

Configure R1 to limit the number of received prefixes from R5 to 10. R1 should be configured to receive a warning message once 50% of this threshold is reached and a warning message for every additional route that exceeds the threshold. You should configure Lo1–Lo10 on R5 by copying and pasting the initial configuration, called **EIGRP-Lab-1-Task10**.

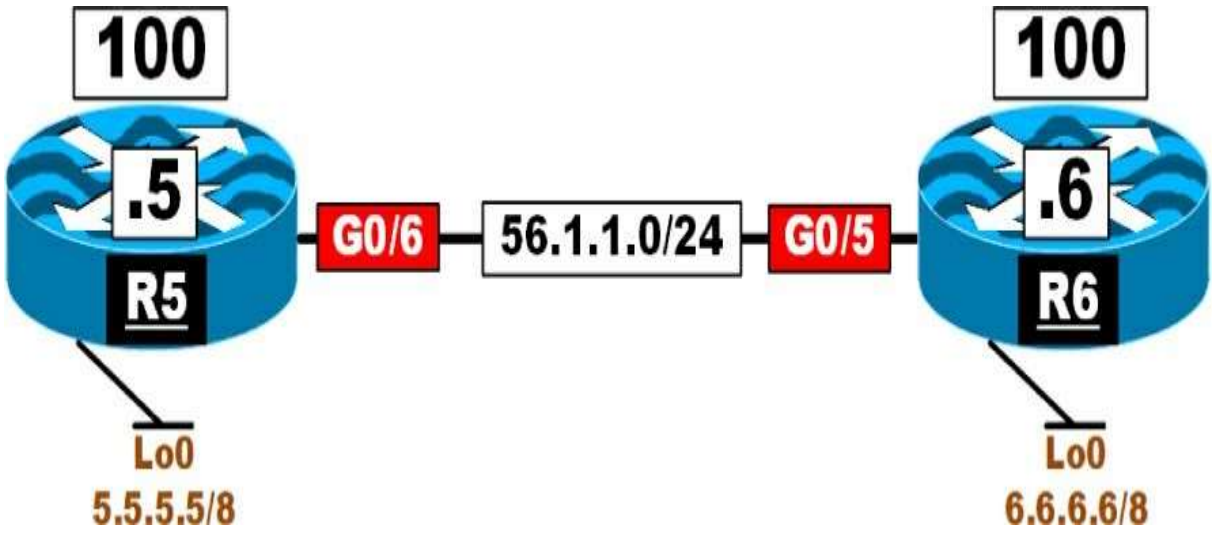
Task 11

Configure R1 to limit the number of prefixes received from R4 to five. R1 should be configured to tear down the adjacency if R4 exceeds the specified threshold. Copy and Paste the **EIGRP-Lab-1-Task11** initial configuration on R4.

Task 12

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 2: EIGRP and BFD



Task 1

Configure the routers based on the previous diagram. *Do not* configure any routing protocol.

Task 2

Configure EIGRP AS 100 on all directly connected interfaces of these two routers and ensure reachability. R5 should be configured in classical mode, and R6 in named mode.

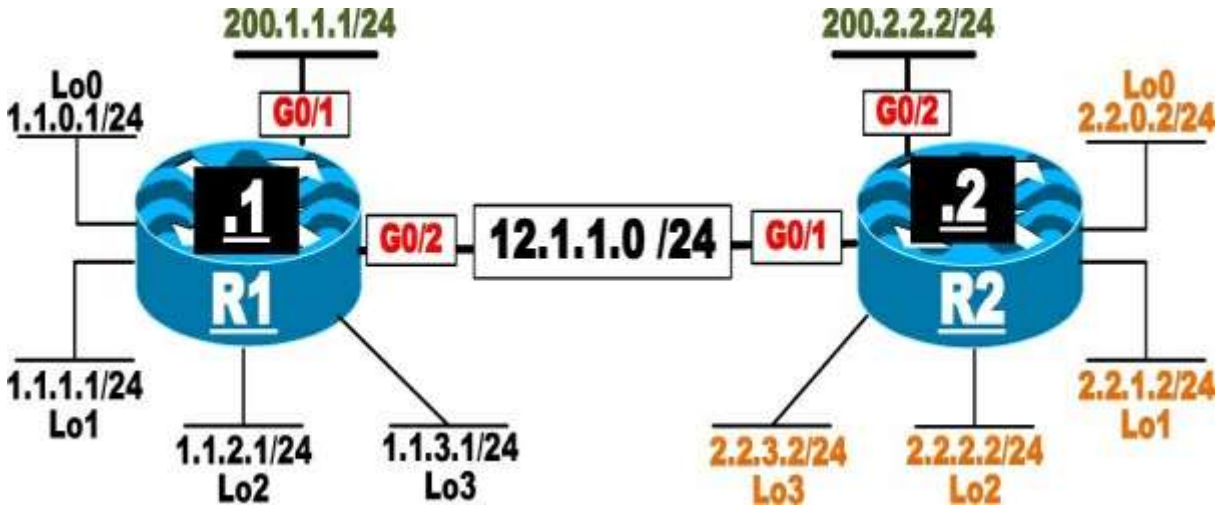
Task 3

Configure and test BFD on these two routers.

Task 4

Erase the startup configuration of these two routers and reload the devices before proceeding to the next lab.

Lab 3: EIGRP Stub



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 3-EIGRP Stub in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-3.

Task 1

Configure EIGRP AS 100 on the G0/2 and G0/1 interfaces of R1 and R2, respectively, as well as on all loopback interfaces of these two routers. On R1 configure EIGRP using the classic mode, and on R2 configure EIGRP in named mode to accomplish this task. ***Do not*** **run EIGRP on the G0/1 interface of R1 or the G0/2 interface of R2.**

Task 2

Configure R1 and R2 to summarize their loopback interfaces in EIGRP.

Task 3

Configure the following static routes on R1 and R2 and redistribute them into EIGRP:

- On R1: 11.0.0.0/8 via G0/1
- On R2: 22.0.0.0/8 via G0/2

Task 4

Advertise the G0/1 interface of R1 and the G0/2 interface of R2 in RIPv2 and disable auto-summarization. You should redistribute RIP into EIGRP and use any metric for redistributed routes.

Task 5

Configure EIGRP stub routing on R1 by using the command **eigrp stub connected**. Test this option and verify the routes in the routing tables of both routers.

Task 6

Remove the **eigrp stub connected** option configured in the previous task and reconfigure EIGRP stub routing on R1 by using the **eigrp stub summary** command. Test this option and verify the routes in the routing tables of both routers.

Task 7

Remove the **eigrp stub summary** option configured in the previous task and reconfigure EIGRP stub routing on R1 by using the command **eigrp stub Static**. Test this option and verify the routes in the routing tables of both routers.

Task 8

Remove the **eigrp stub static** option configured in the previous task and reconfigure EIGRP stub routing on R1 by using the command **eigrp stub redistributed**. Test this option and verify the routes in the routing tables of both routers.

Task 9

Remove the **eigrp stub redistributed** option configured in the previous task and reconfigure EIGRP stub routing on R1 by using the command **eigrp stub receive-only**. Test this option and verify the routes in the routing tables of both routers.

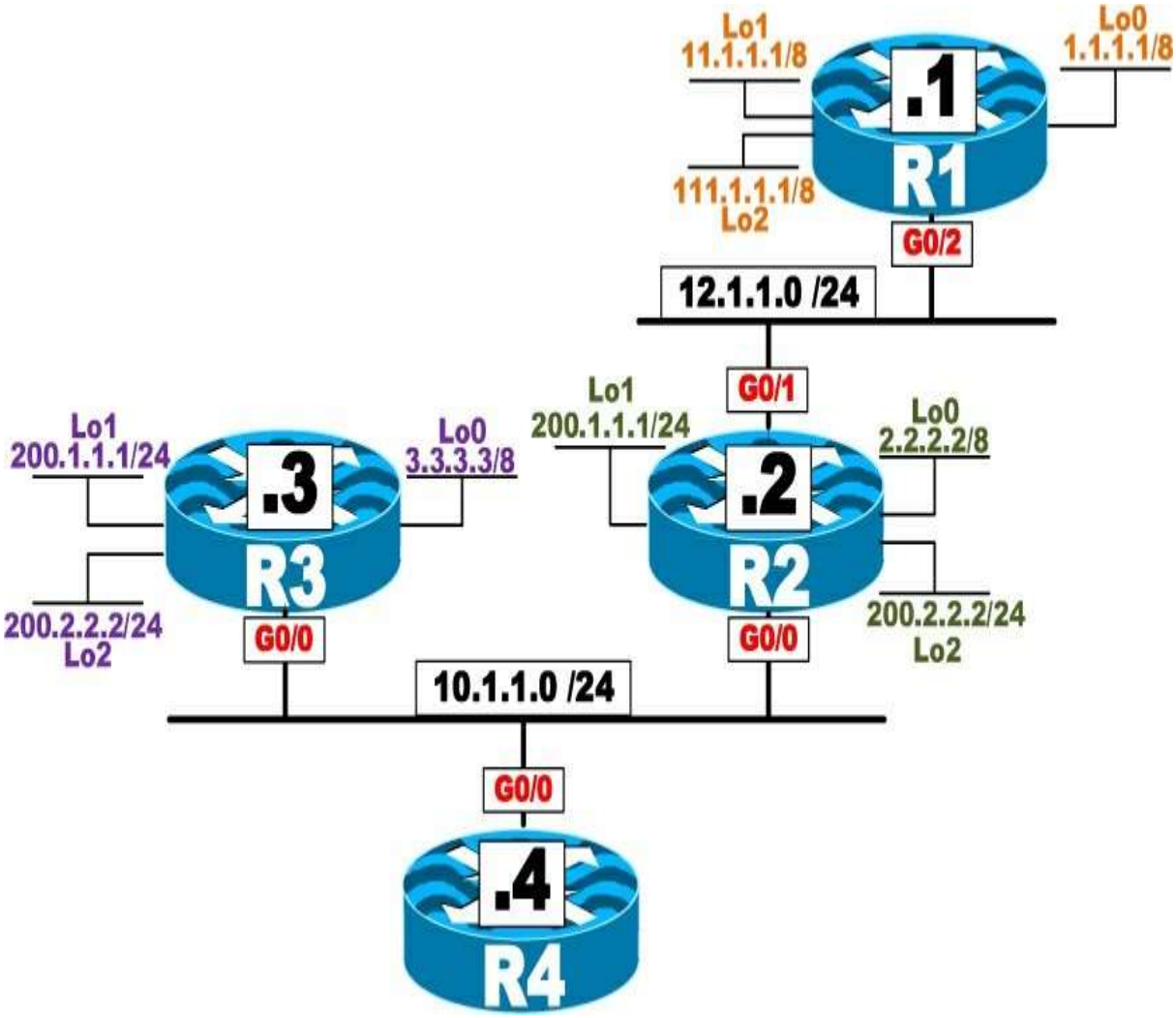
Task 10

Remove the **eigrp stub receive-only** option configured in the previous task and reconfigure EIGRP stub routing on R1 by using the command **eigrp stub**. Test this option and verify the routes in the routing tables of both routers.

Task 11

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 4: EIGRP Filtering



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 4-EIGRP Filtering in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-4.

Task 1

Configure EIGRP 100 on all routers and advertise their directly connected links.

Task 2

Configure R4 such that it filters existing (1.0.0.0/8, 11.0.0.0/8, and 111.0.0.0/8) and future networks behind R1. *Do not* use **distribute-list**, **access-list**, **prefix-list**, or **route-map** to accomplish this task

Task 3

Configure R4 such that it uses R2 as its primary connection to network 200.1.1.0 /24. You should use an access list to accomplish this task.

Task 4

Configure R4 such that it takes R3 to reach network 200.2.2.0 /24. R4 should *only* use R2 as the next hop to reach network 200.2.2.0/24 when R3 is down. You should use a standard access list to accomplish this task.

Task 5

Filter network 2.0.0.0/8 on R4. *Do not* use **distribute-list** or **route-map** to accomplish this task.

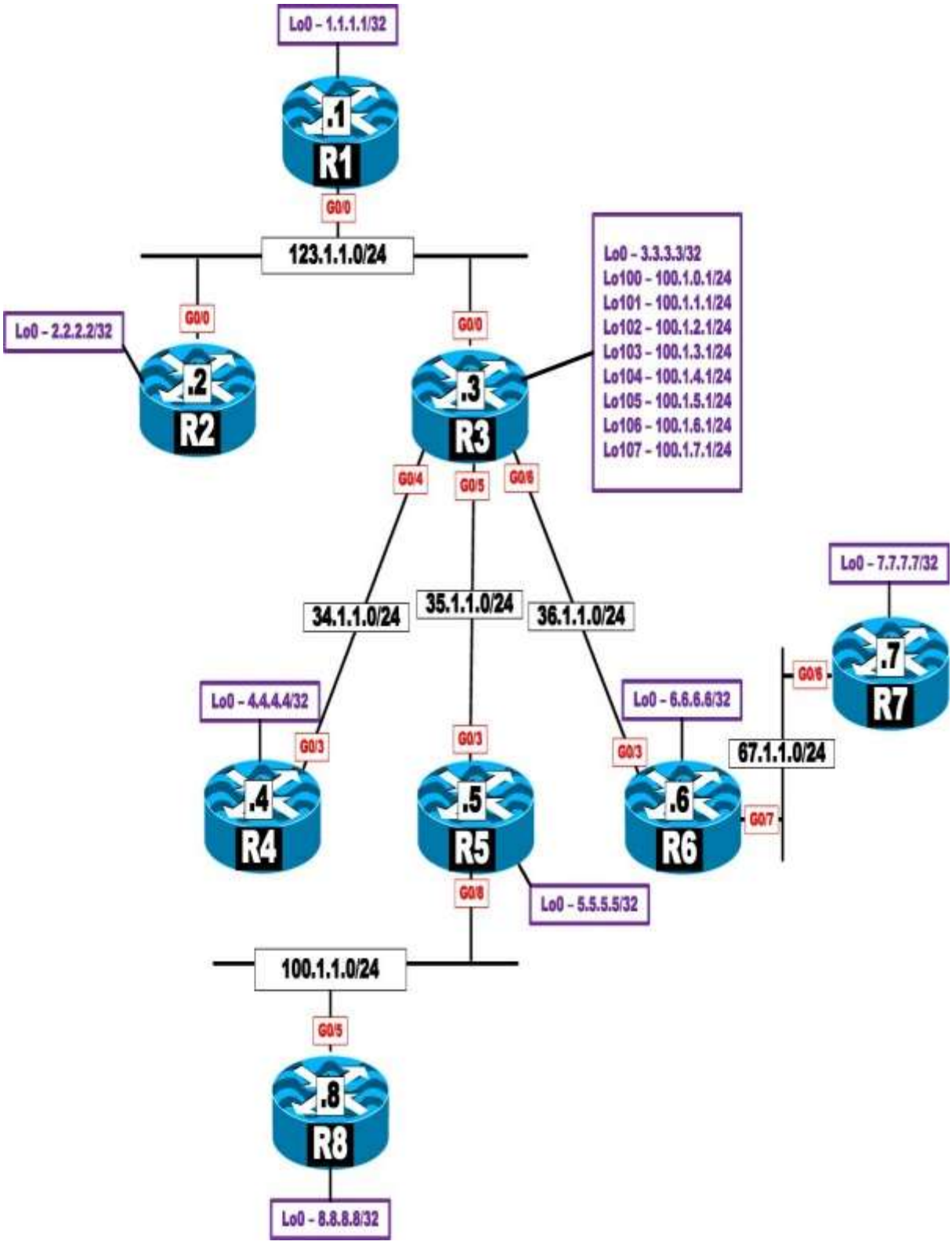
Task 6

Configure R4 to filter network 3.0.0.0/8.

Task 7

Erase the startup configuration and reload the routers before proceeding to the next task.

Lab 5: Advanced EIGRP Lab


Lo0 - 1.1.1.1/32
.1
R1
G0/0
123.1.1.0/24
Lo0 - 2.2.2.2/32
G0/0
.2
R2
G0/0
.3
R3
Lo0 - 3.3.3.3/32
Lo100 - 100.1.0.1/24
Lo101 - 100.1.1.1/24
Lo102 - 100.1.2.1/24
Lo103 - 100.1.3.1/24
Lo104 - 100.1.4.1/24
Lo105 - 100.1.5.1/24
Lo106 - 100.1.6.1/24
Lo107 - 100.1.7.1/24
G0/4
G0/5
G0/6
34.1.1.0/24
35.1.1.0/24
36.1.1.0/24
Lo0 - 7.7.7.7/32
.7
R7
G0/6
Lo0 - 4.4.4.4/32
G0/3
.4
R4
G0/3
.5
R5
G0/3
Lo0 - 6.6.6.6/32
.6
R6
67.1.1.0/24
G0/7
G0/0
Lo0 - 5.5.5.5/32
100.1.1.0/24
G0/5
.8
R8
Lo0 - 8.8.8.8/32

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 5-Advanced EIGRP Lab in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-5.

Task 1

Configure the G0/0 interfaces of R1, R2, and R3 in EIGRP AS 100. These routers should be configured to advertise their Lo0 interfaces in this AS, using the following policy:

- These routers should be configured to reach each other's loopback interface/s by going through R1.
- *Do not* use PBR or configure another AS to accomplish this task.

Task 2

Configure R3's G0/4, G0/5, and G0/6 in AS 300. Configure R4's, R5's, and R6's G0/3 and lo0 interfaces in this AS.

Configure R3 to summarize its Lo100–Lo107.

The summary route should be advertised to R4, R5, and R6 based on the following policy:

- R4 should receive the summary *only*.
- R5 should receive the summary plus network 100.1.3.0 /24.
- R6 should receive the summary plus all the specific routes.
- Configure the minimum number of **ip summary-address** commands possible to accomplish this task.

Task 3

Configure EIGRP 300 on R4's Lo134 and Lo135 and advertise a single summary in AS 300.

Task 4

Configure the G0/7 and Lo0 interfaces of R6 and the G0/6 and loopback 0 interfaces on R7 for EIGRP in AS 67

R7 should be configured to advertise its Lo130, such that the command **show ip route eigrp 67** on R6 resembles the following:

```
D EX  130.3.0.0/16 [170/130816] via 67.1.1.7,
00:00:16, GigabitEthernet0/7
```

R7 should use **redistribute static** to accomplish this task. *Do not* configure a static route to accomplish this task.

Task 5

Configure the routers in AS 67 such that they log neighbor warning messages and repeat the warning messages every 10 minutes. You should disable logging of neighbor changes for this AS.

Task 6

Configure the routers in AS 67 such that a dead neighbor is detected within 3 seconds.

Task 7

Routers in AS 100 should be configured to use **Bandwidth** and *not* **Bandwidth + DLY** when calculating their composite metric.

Task 8

Configure R2 such that EIGRP *never* uses more than 25% of its G0/0 link's bandwidth.

Task 9

Configure the G0/8 interface of R5 and the G0/5 and the Lo0 interfaces of R8 in AS 500.

Task 10

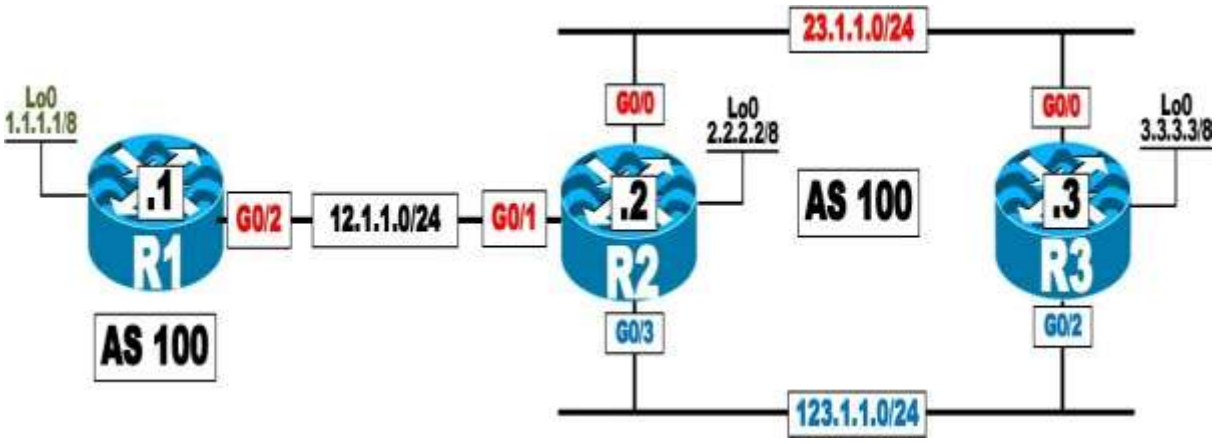
Configure R5 to inject a default route in AS 500 based on the following policy:

- R5 should be configured to inject a default route plus networks 4.0.0.0/8 and 6.0.0.0/8 from AS 300.

Task 11

Erase the startup configuration and reload the routers before proceeding to the next task.

Lab 6: EIGRP Authentication



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 6-EIGRP Authentication in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-6.

Task 1

Configure EIGRP based on the previous diagram. If this configuration is successful, these routers should be able to see and have reachability to all routes. You should use named mode when configuring R2 and R3 and classic EIGRP configuration when configuring R1 to accomplish this task.

Task 2

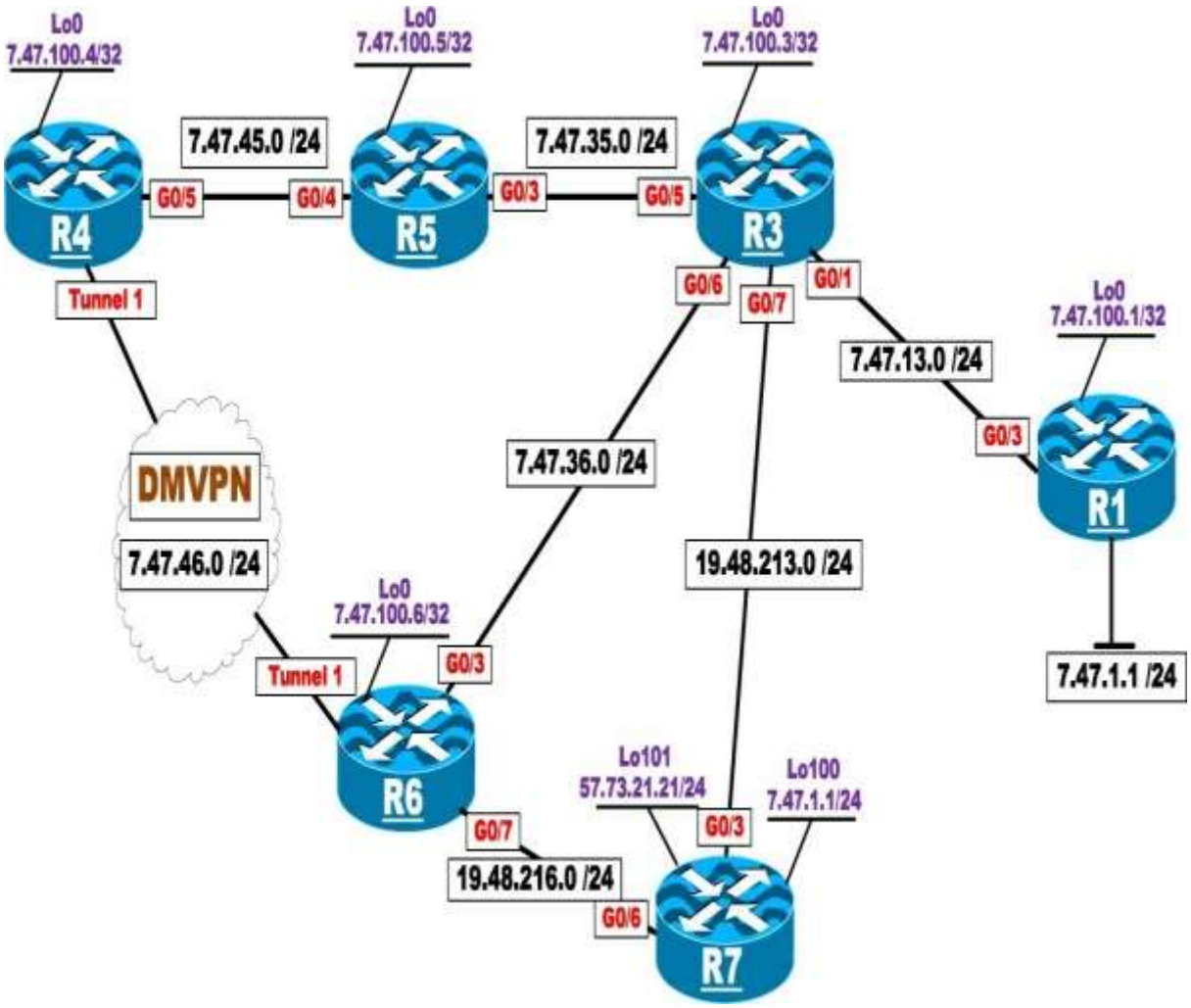
Configure R2 to authenticate all existing and future directly connected interfaces using the strongest authentication method available. Use the minimum number of commands and **CCIE** as the password to accomplish this task.

- R2 should authenticate R1 using MD5 and **Cisco** as the password.
- In the future, R3 may have other neighbors that won't need authentication.

Task 3

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 7: EIGRP Challenge Lab



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 7-EIGRP Challenge Lab in the EIGRP folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → EIGRP folder → Lab-7.

Note

Do not access R7 at all. You should *only* fix the problem identified in the ticket.

Ticket 1

R1 can't reach R3's Lo0. You must configure R1 to fix the problem.

Ticket 2

R6 does *not* have a stable EIGRP adjacency with R4. *Do not* use an EIGRP command to fix this ticket.

Ticket 3

When R3's G0/1, G0/7, and G0/6 are down, R3 can't reach R4's Lo0. *Do not* remove any commands to fix this ticket.

Ticket 4

R1's Lo0 should always have reachability to R4's Lo0 and G0/5 interfaces, but it does not. You should fix this problem without configuring R1 or R4. You should *not* remove any commands to resolve this ticket.

Ticket 5

R3 is configured to use multiple paths to R4's Lo0. However, it's using only one of the paths.

Ticket 6

R6 can’t reach R7’s Lo101.

Ticket 7

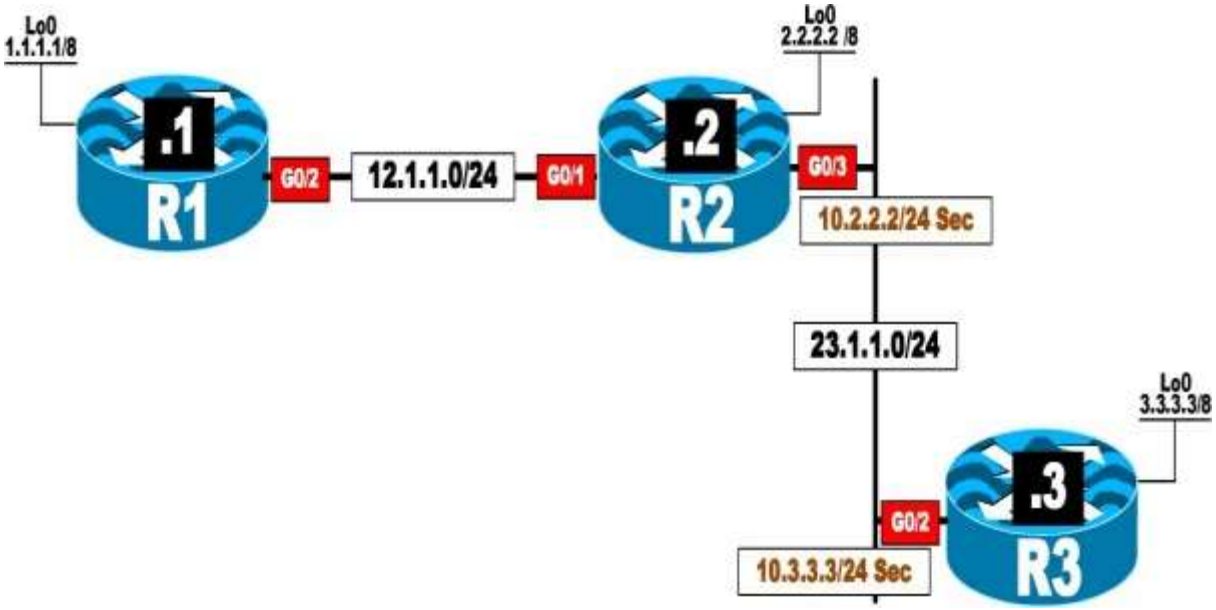
R3 should establish a EIGRP adjacency with R8 over its G0/8 interface. You should make configuration changes on R3 only.

Ticket 8

Erase the startup configuration and reload the devices before proceeding to the next lab.

Chapter 6. OSPF

Lab 1: Running OSPF on the Interfaces



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 1-Running OSPF on the interfaces in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-1.

Task 1

Configure OSPF Area 0 on all directly connected interfaces in the previous topology, including the secondary IP addresses.

- Do not use the **network** command to accomplish this task.
- The loopback interfaces should be advertised with their correct mask.
- If this configuration is performed successfully, you should have reachability to all routes within the previous topology.
- The OSPF RID should be configured based on the Lo0 interface ofthese three routers.

Task 2

Configure R2 and R3 such that the secondary IP addresses are not advertised.

- *Do not* use filtering, a route map, an access list, or a prefix list.
- *Do not* remove any commands.
- Use a minimum number of commands to accomplish this task.

Task 3

R3 is getting flooded with LSA Type 6 packets. Ensure that R3 does not generate a syslog message for this LSA type.

Task 4

To ensure fast detection of a neighbor being down, configure R2 and R3 to send their hellos every 250 milliseconds with a hold time of 1 second for their Ethernet link.

Task 5

Ensure that these routers look up DNS names for use in most of the OSPF **show** commands. Test this task to ensure proper operation. Since there are no DNS servers in this lab, you should use the local routers for DNS lookups.

Task 6

Configure R2 such that if it does not receive an acknowledgment from R3 for a given LSA, it waits 10 seconds before it re-sends that given LSA.

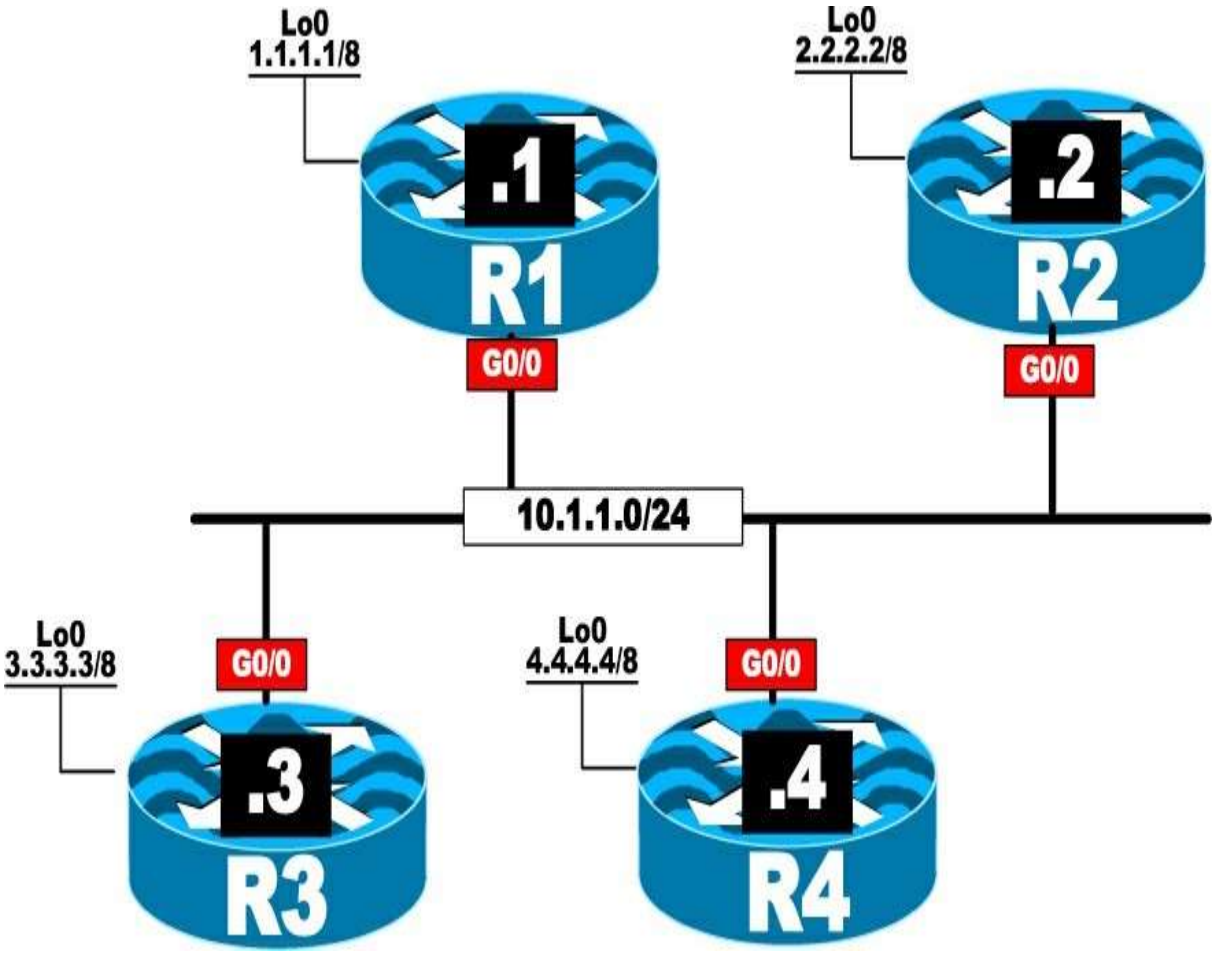
Task 7

Configure R1 and R2 such that when there is a topology change in Area 0 for LSA Types 1 and 2, the entire SPT is *not* recomputed. This should *only* occur for the affected part/s of the tree.

Task 8

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 2: OSPF Broadcast Networks



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 2-OSPF Broadcast Network in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-2.

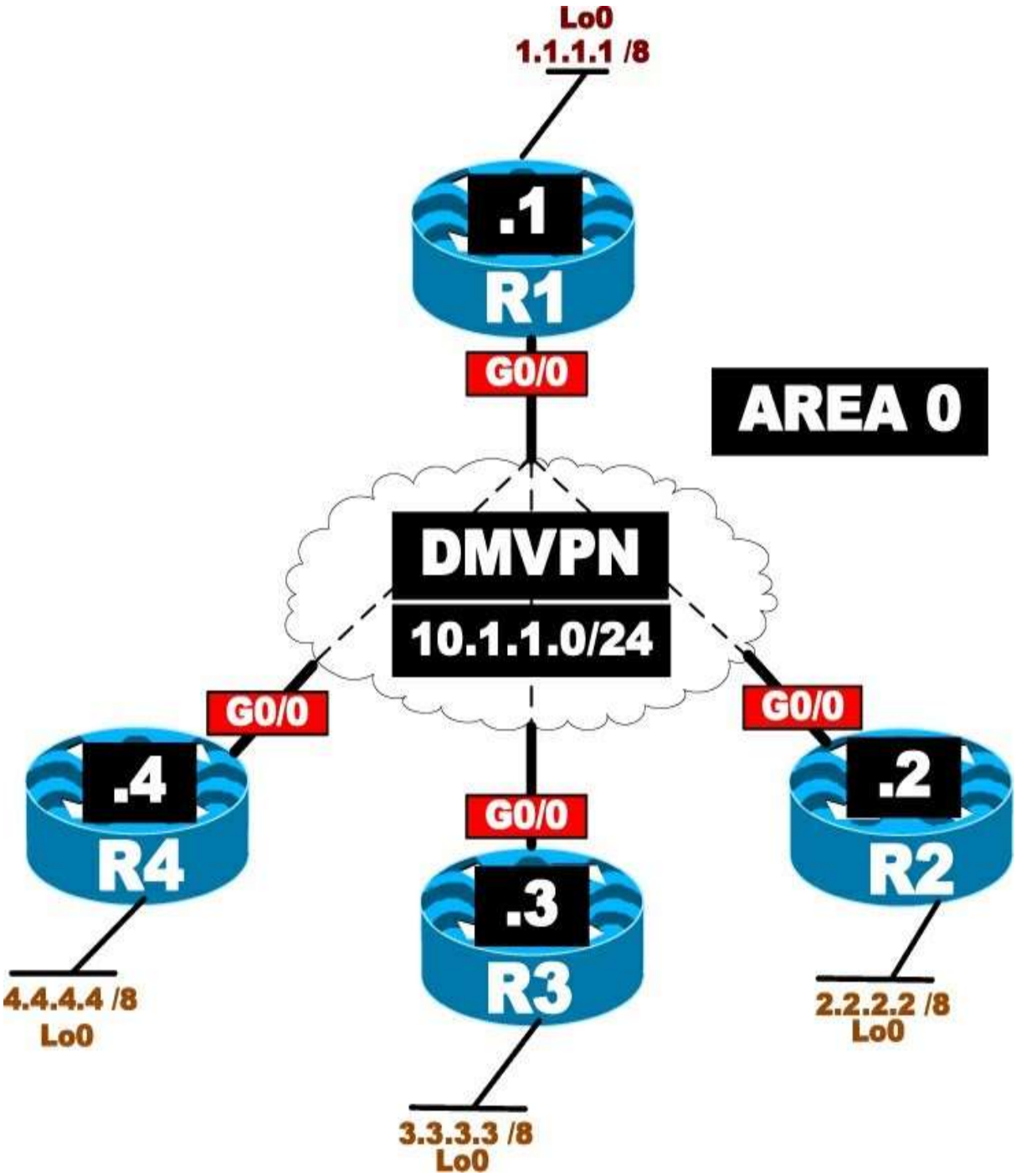
Task 1

Configure OSPF Area 0 on all routers and run their directly connected interfaces. Ensure that Loopback0 interfaces are advertised with their correct mask. You should configure the OSPF router IDs to be 0.0.0.*x*, where *x* is the router number.

Task 2

Lab Setup:

Erase the configuration of all routers and SW1. To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-2-Task-2.



Configure OSPF on the tunnel and Loopback0 interfaces of all routers, based on the following policy:

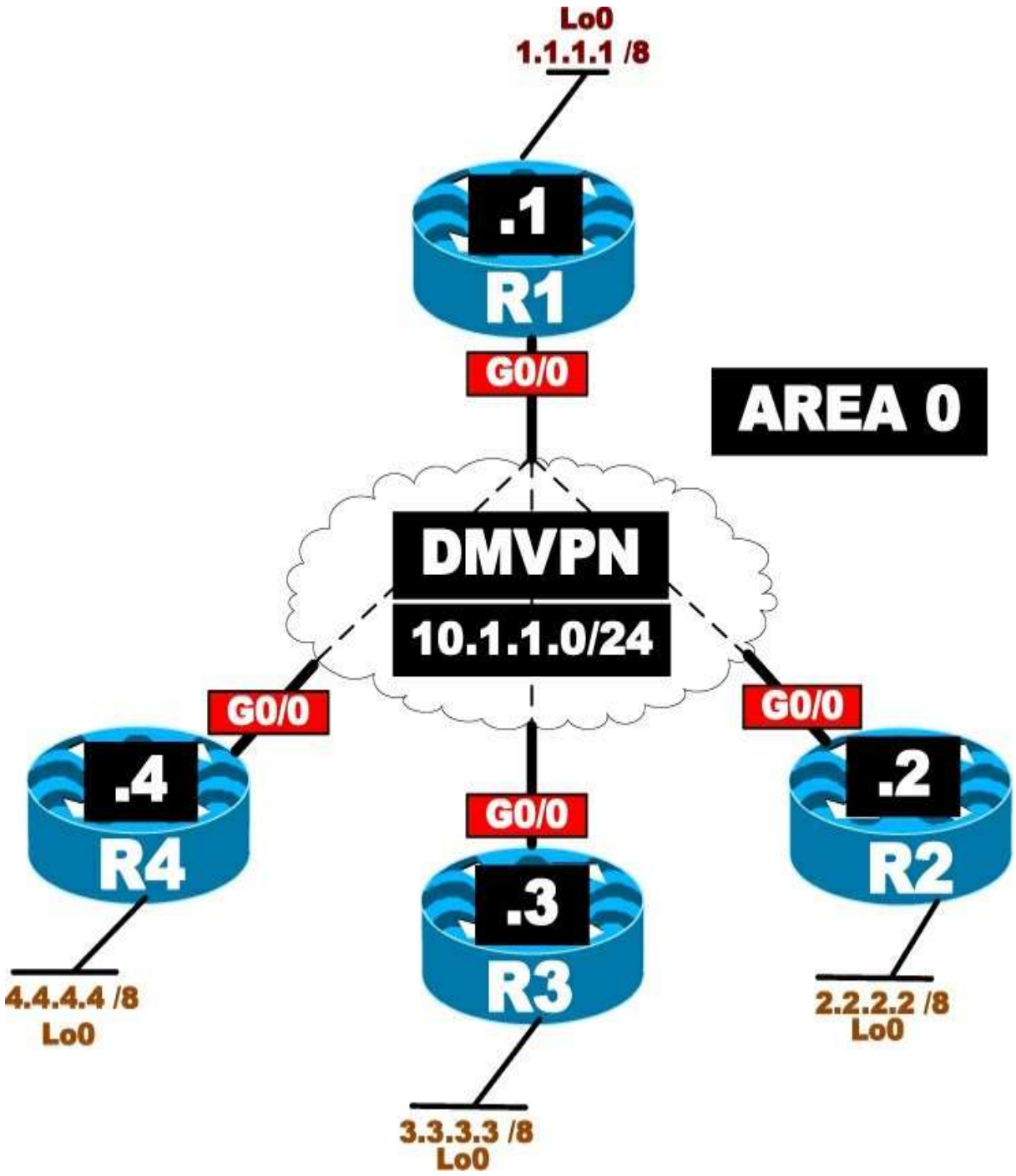
- R1 is the hub, and R2, R3, and R4 are configured as spokes. *Do not* change the topology.

- Configure the tunnel interfaces of all routers to be OSPF broadcast network type.
- The loopback interfaces should be advertised with their correct mask.
- Configure the router IDs of R1, R2, R3, and R4 to be 0.0.0.1, 0.0.0.2, 0.0.0.3, and 0.0.0.4, respectively.
- You should use static maps on the DMVPN network.• Spokes should traverse the hub to reach all networks.

Task 3

Erase the startup configuration of the routers, the config.text file, and the VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 3: OSPF Non-broadcast Networks



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 3-OSPF Non-Broadcast Networks in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-3.

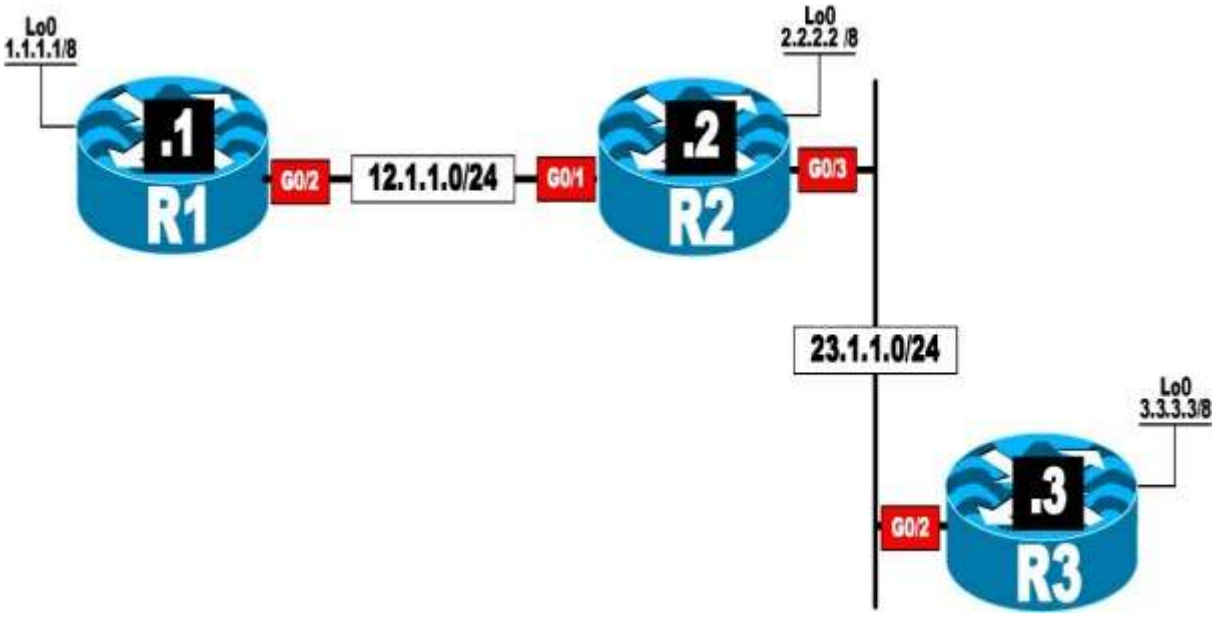
Task 1

Configure OSPF Area 0 on all routers and their directly connected interfaces. You should configure the tunnel interfaces as the OSPF non-broadcast network type. Configure the OSPF router IDs to be 0.0.0.*x*, where *x* is the router number.

Task 2

Erase the startup configuration of the routers, the config.text file, and the VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 4: OSPF Point-to-Point Networks



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 4-OSPF Point-to-Point Networks in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-4.

Task 1

Configure OSPF on all routers and run their directly connected interfaces in Area 0, based on the following policy:

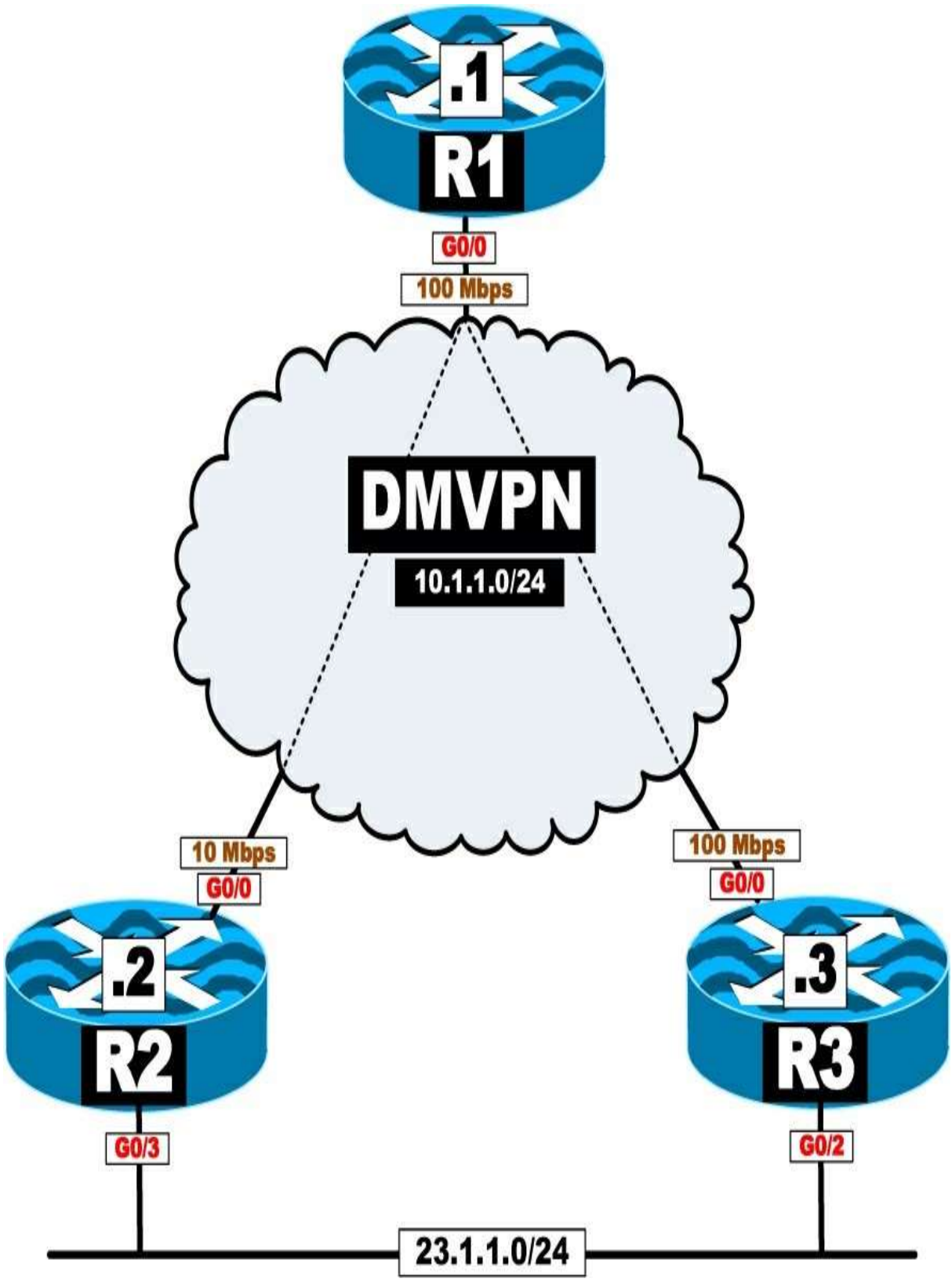
- The Loopback0 interfaces of these routers should be advertised withtheir correct mask.
- Use 0.0.0.1, 0.0.0.2, and 0.0.0.3 as the router IDs of R1, R2, and R3,respectively.

- There should *not* be any DR/BDR election on any of the links.
- *Do not* configure point-to-multipoint or point-to-multipoint nonbroadcast on any of the links.

Task 2

Erase the startup configuration of the routers, the config.text file, and the VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 5: OSPF Point-to-Multipoint and Point-to-Multipoint Nonbroadcast Networks



This lab should be conducted on the Enterprise

POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 5-OSPF Point-to-Multipoint & Point-to-Multipoint non-broadcast Networks in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-5.

Task 1

Configure OSPF Area 0 on all links in the previous topology. The OSPF router IDs of all routers should be 0.0.0.*x*, where *x* is the router number. If this configuration is performed successfully, the routers in this topology should have full NLRI to every network in this topology. The tunnel interface of these routers should not perform DR/BDR election.

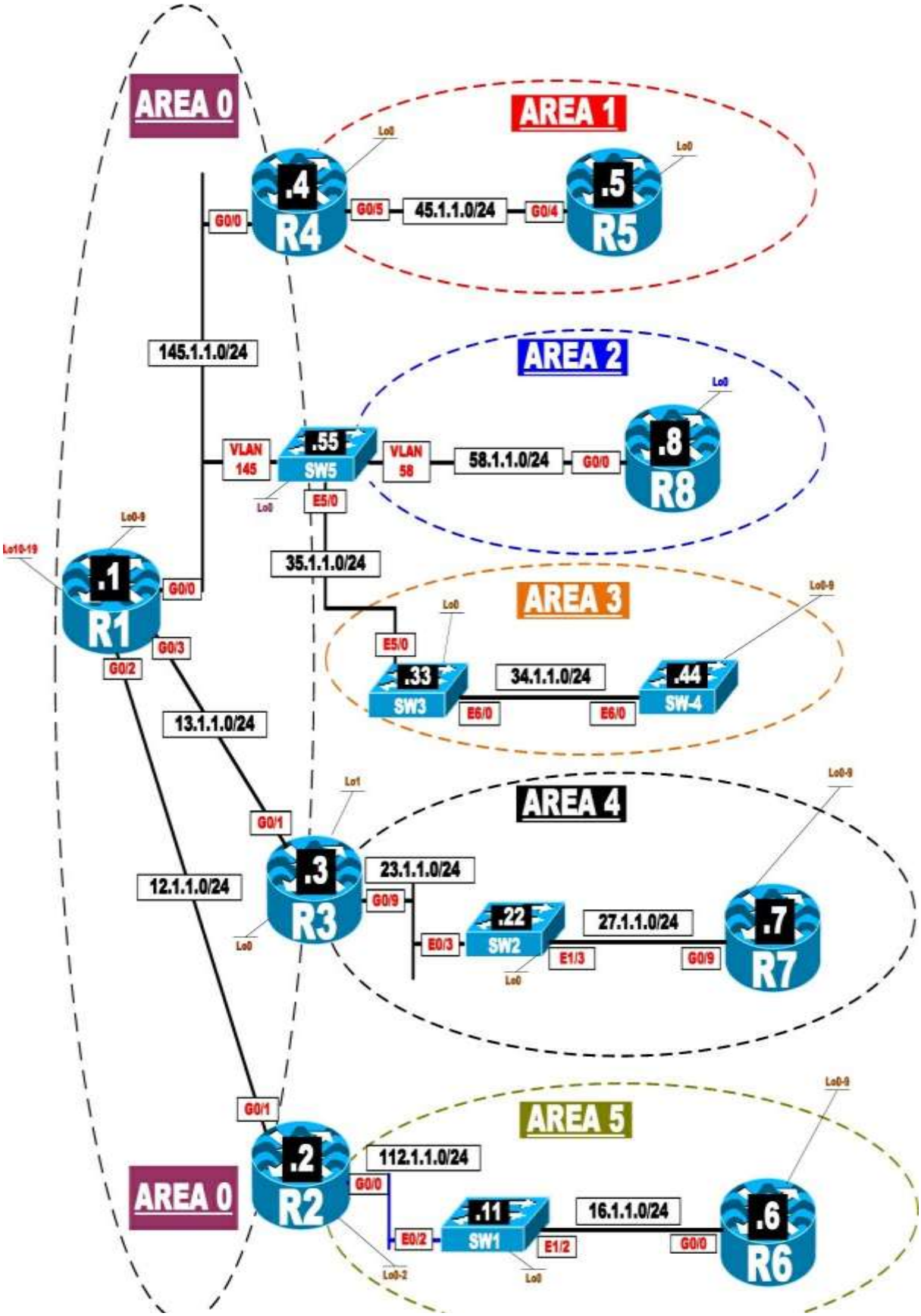
Task 2

Since R2's connection to the cloud is 10 Mbps and R3's connection is 100, R1 should *not* perform equal-cost load sharing. R1 should go through R3 to reach network 23.1.1.0/24. *Do not* configure **PBR** or the **IP ospf cost** command to accomplish this task.

Task 3

Erase the startup configuration of the routers, the config.text file, and the VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 6: OSPF Area Types



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 6-OSPF Area Types in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-6.

Lab rules:

- All loopback interfaces are configured with **ip ospf network point-topoint**.
- Configure the OSPF router IDs of the routers based on the followingchart:

R1: 0.0.0.1	R2: 0.0.0.2	R3: 0.0.0.3
R4: 0.0.0.4	R5: 0.0.0.5	R6: 0.0.0.6
R7: 0.0.0.7	R8: 0.0.0.8	SW1: 0.0.0.11
SW2: 0.0.0.22	SW3: 0.0.0.33	SW4: 0.0.0.44
SW5: 0.0.0.55		

Task 1

Configure the Lo0 and G0/4 interfaces of R5 and G0/5 and the Lo0 interface of R4 in Area 1.

Task 2

Configure the G0/0 and Lo0 interfaces of R8 and the VLAN58 interface of SW5 in Area 2.

Task 3

Configure the following interfaces for OSPF Area 3:

- E6/0 interface on SW4
- E6/0, E5/0, and Loopback0 interfaces on SW3
- E5/0 interface on SW5

Ensure that SW4 is configured to redistribute its Loopback0 through Loopback9 interfaces into this routing domain.

Task 4

Configure the following interfaces in Area 4:

1. R7's G0/9 interface
2. SW2's Lo0, e1/3, and e0/3 interfaces
3. R3's G0/9 interface

Ensure that R7 is configured to redistribute the networks on its Loopback0 through Loopback9 interfaces into this routing domain.

Task 5

Configure the following interfaces in Area 5:

- R6's G0/0 interface
- SW1's Lo0, e1/2, and e0/2 interfaces
- R2's Loopback0 through Loopback2 and G0/0 interfaces

Ensure that R6 is configured to redistribute its Loopback0 through Loopback9 interfaces into this routing domain.

Task 6

Configure the following interfaces in Area 0:

- R1's Loopback0 through Loopback9, G0/0, G0/3, and G0/2 interfaces
- R1 must be configured to redistribute its Lo10–Lo19 in this routingdomain.
- R4's G0/0 interface
- SW5's VLAN145 and Loopback0 interfaces
- R3's G0/1 and Loopback 0 interfaces
- R2's G0/1 interface

Restrictions:

- R4 should *not* use a **network** command to run its G0/0 interface in Area 0.
- This router should *not* advertise any secondary IP address/es configured under its G0/0 interface.
- *Do not* configure any kind of filtering to accomplish this task.

Task 7

Ensure that the routers in Area 1 do not have any external routes in their routing table; these routers should not have LSA Types 4 or 5 in their LSDB.

Task 8

Configure Area 2 such that existing and future external and inter-area routes are *never* seen in the routing tables of these routers. These routers should *not* have LSA Types 3, 4, or 5 in their LSDB, but these routers should have full reachability to the inter-area and external networks redistributed in this routing domain.

Restriction:

- *Do not* use an ACL or a prefix list to accomplish this task.

Task 9

Configure the routers in Area 3 based on the following policy:

- The routers must maintain existing and future inter-area routes in theirrouting tables.
- The routers *should not* have LSA Types 4 or 5 in their OSPF link state database.
- The routers *should not* have reachability to the routes redistributed in the other areas of this routing domain.
- The routers should have reachability to the routes redistributed in theirown area.

Task 10

Configure the routers in Area 4 based on the following policy:

- The routers of this area *should* have LSA Type 3 in their OSPF link state database.
- The routers of this area *should not* have LSA Types 4 or 5 in their OSPF link state database.
- The routers *must* have reachability to the existing and future routes redistributed in the other areas of this routing domain, except for the external networks redistributed in Area 3.
- The routers should have reachability to the networks redistributed intheir own area.

Task 11

Implement the following policy for the routers in Area 5:

- The routers must have reachability to the routes redistributed in thisrouting domain.

- The routers *should not* maintain LSA Type 3 in the OSPF link state database.
- The routers should *not* maintain LSA Types 4 or 5 in their OSPF link state database.

Task 12

Determine whether the routers in Area 0 maintain LSA Type 4 in their OSPF link state database?

Task 13

Configure R3 to redistribute its Lo1 interface such that existing and future redistributed routes by this router are *only* injected into Area 0 and *not* into Area 4. *Do not* configure an ACL or a prefix list to accomplish this task.

Task 14

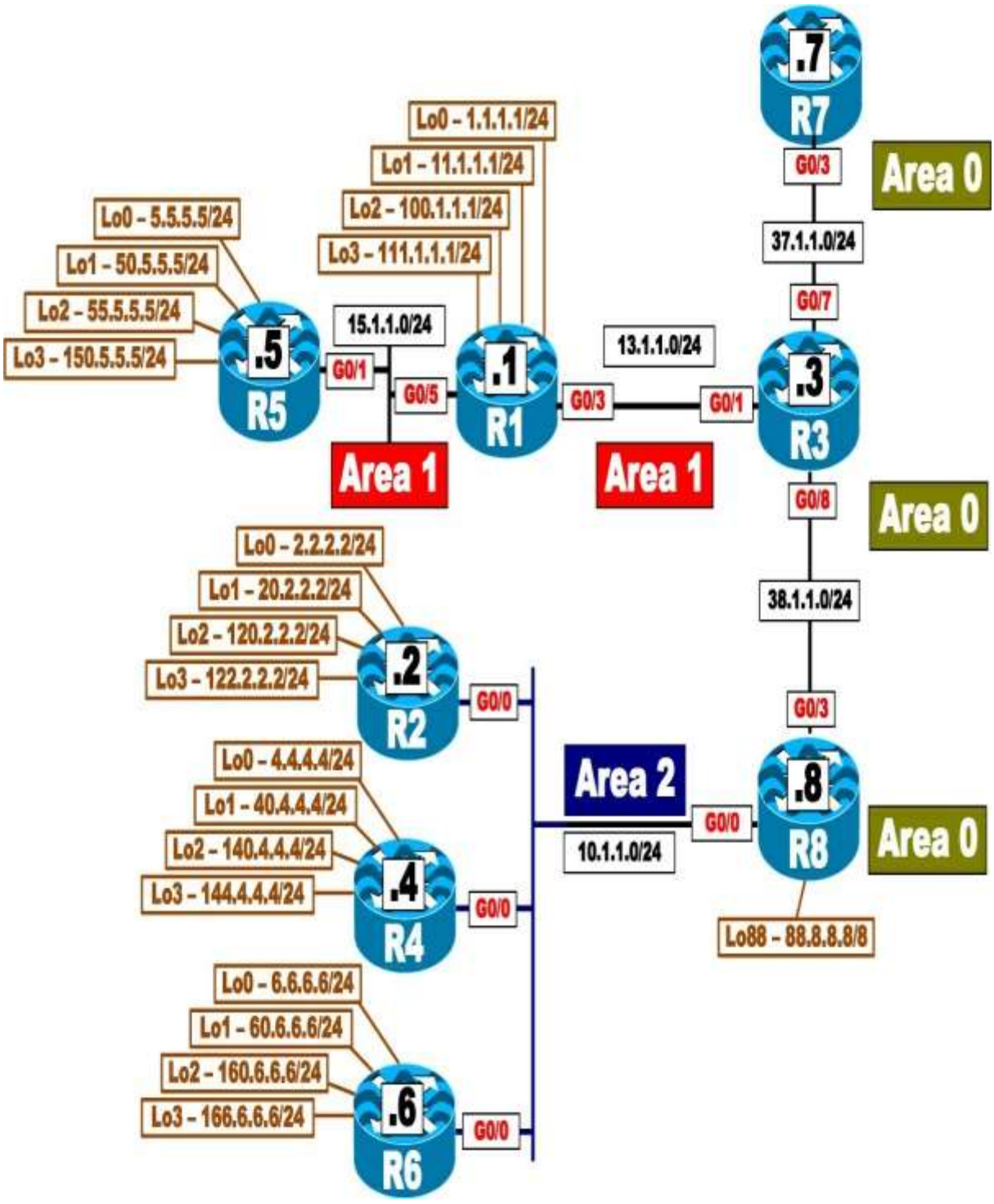
Configure the following ABRs so that the default route that they inject has an OSPF cost based on the following table:

ABR/Area	OSPF Cost of the Injected Default Route
R4/Area 1	40
R9/Area 2	133
R3/Area 4	30
R2/Area 5	20

Task 15

Erase the startup configuration of the routers, the config.text file, and the VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 7: OSPF Filtering



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 7-OSPF Filtering in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-7.

Pre-configuration:

- OSPF is configured on all routers, and all loopback interfaces are configured with their correct masks.
- OSPF router IDs are configured as 0.0.0.*x*, where *x* is the router number.

Task 1

Configure R1 and R3 such that the link connecting them to each other is *not* advertised. R1 and R3 should still maintain their adjacency through this interface.

Task 2

Configure R5 such that it *only* advertises its Lo0 and Lo1. *Do not* remove or modify the **network** command/s configured in the router configuration mode. You should use two different solutions: one for the Lo0 interface and a second one for the Lo1 interface.

Task 3

Configure R3 to redistribute all OSPF prefixes into BGP AS 100. R8 should redistribute its Lo88 into OSPF interfaces such that R7 filters network 88.0.0.0 /8 from getting into its BGP table.

Task 4

Configure LSA Type 3 filtering on R3 to filter network 1.1.1.0 /24 from the rest of the OSPF domain. You should reference Area 1 when accomplishing this task.

Task 5

Configure LSA Type 3 filtering on R8 to filter network 5.5.5.0 /24 from getting into Area 2.

Task 6

Configure LSA Type 3 filtering on R8 to filter network 50.5.5.0 /24. You should reference Area 0 when accomplishing this task.

Task 7

Configure R8 such that network 100.1.1.0 /24 is *not* advertised to the routers in Area 2. *Do not* use the following to accomplish this task:

- A distribute list, an area filter list, distance, a route map, an access list,or a prefix list

Task 8

Configure R3 such that network 111.1.1.0 /24 is *not* advertised to routers in Area 0 or Area 2. *Do not* add any static routes, use a filter list, or use a distribute list to accomplish this task.

Task 9

Enable OSPF Area 2 on the G0/9 interfaces of R2, R4, and R6.

Task 10

Configure R2, R4, and R6 based on the following policy:

- R2 should redistribute networks 120.2.2.0 /24 and 122.2.2.0 /24 as OSPF external Type 1 networks.
- R2 should redistribute its Lo0 and Lo1 interfaces into the OSPF routingdomain.
- R4 should redistribute networks 140.4.4.0 /24 and 144.4.4.0 /24 as OSPF external Type 1 networks.
- R4 should redistribute its Lo0 and Lo1 interfaces into the OSPF routingdomain.

Task 11

Configure R2 and R6 to redistribute their Lo0 interfaces. Configure the appropriate routers such that the routers in Area 2 can see networks 2.2.2.0 /24, 4.4.4.0 /24, and 6.6.6.0 /24 in their routing tables. The routers in the other areas, however, should *not* have these networks in their routing table.

Task 12

Configure R2 to filter network 122.2.2.0 /24. The other routers should *not* have this route in their routing table or database.

- *Do not* use **summary-address** to accomplish this task.
- *Do not* modify the redistribution parameters to accomplish this task.

Task 13

Configure R2 to filter existing and future inter-area and/or intra-area routes from its routing table. Use the smallest number of commands possible to accomplish this task.

Task 14

Configure R4 to filter existing and future routes that have an OSPF cost of 20 from its routing table.

Task 15

Configure R6 to filter the default route injected by the ABR by R8 from its routing table.

Task 16

Configure R5 to filter network 1.1.1.0 /24. *Do not* use **distribute-list** to accomplish this task.

Task 17

Configure R1 to filter existing and future external routes. *Do not* configure an access list, a route map, or a prefix list to accomplish this task.

Task 18

Configure R7 to filter existing and future intra-area routes. *Do not* configure a route map, an access list, or a prefix list to accomplish this task.

Task 19

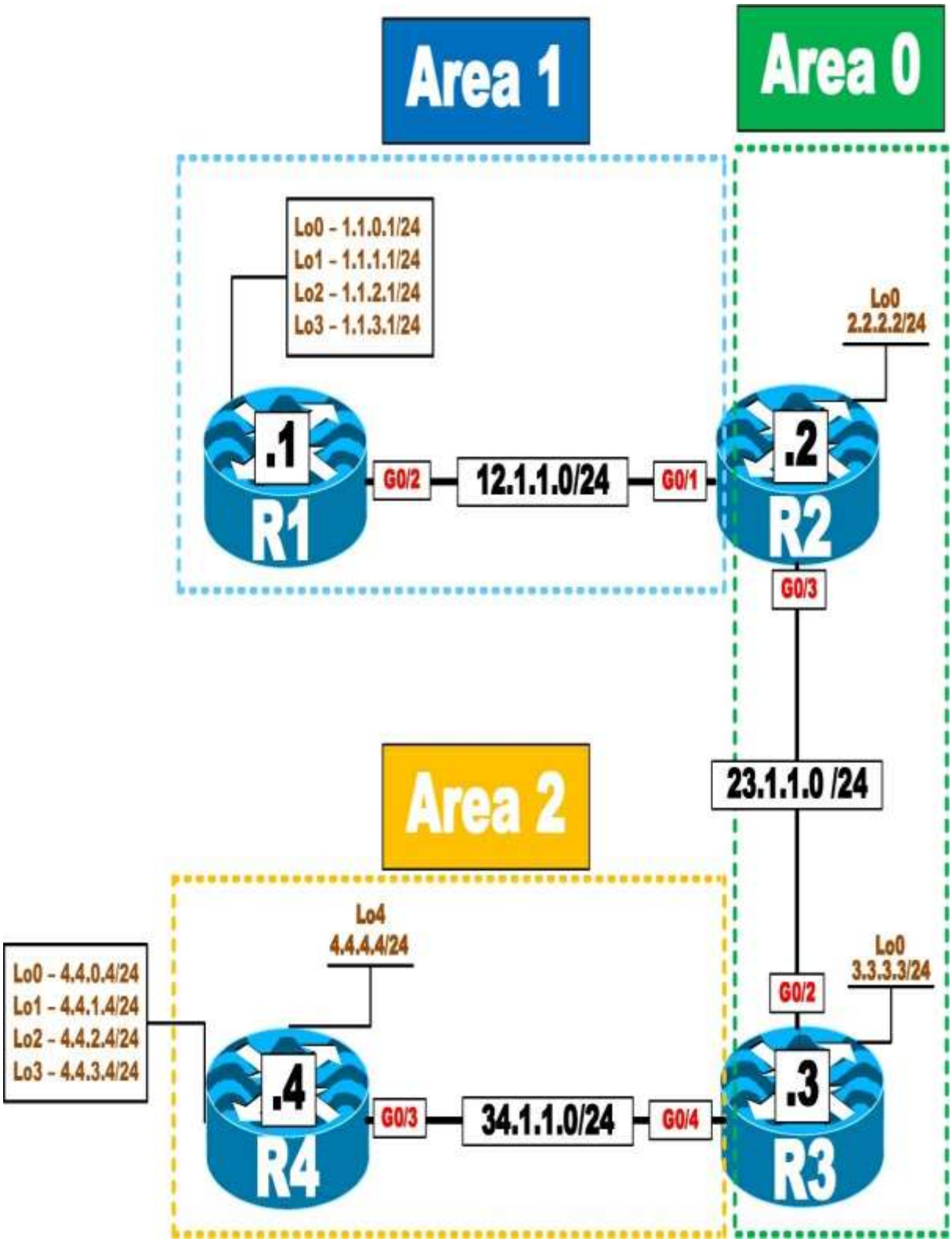
Configure R5 to filter existing and future intra-area routes. *Do not* configure a route map, an access list, or a prefix list to accomplish this task.

Task 20

Erase the startup configuration of the routers, the config.text file, and the

VLAN.dat file of each switch and reload the devices before proceeding to the next lab.

Lab 8: OSPF Summarization



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 8-OSPF Summarization in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-8.

Task 1

Configure the routers as follows:

- R4 should redistribute the four loopback interfaces (4.4.0.4 /24–4.4.3.4/24) into the OSPF routing domain.
- R4 should advertise its Loopback4 and G0/3 interfaces in Area 2.
- R1 should advertise all of its interfaces in OSPF Area 1. Use a minimum number of commands to accomplish this.
- R2 should advertise its Loopback0 and G0/3 interfaces in Area 0. Itshould advertise its G0/1 interface in Area 1.
- R3 should advertise its Loopback0 and G0/2 interfaces in Area 0. Itshould advertise its G0/4 interface in Area 2.
- R1 should use 0.0.0.1, R2 should use 0.0.0.2, R3 should use 0.0.0.3,and R4 should use 0.0.0.4 as their OSPF router IDs.

Task 2

Configure the appropriate router in Area 2 to summarize all external (E2) routes.

Task 3

Configure the appropriate router in Area 1 to summarize the following four networks and only advertise a single summary route:

- 1.1.0.0 /24
- 1.1.1.0 /24
- 1.1.2.0 /24
- 1.1.3.0 /24

Task 4

Ensure that the routers do *not* install a null 0 route in the routing table when they summarize routes. You should show two ways to accomplish this task.

Task 5

Reconfigure Area 2 such that the routers within Area 2 see all the specific routes, but the routers in Area 0 and Area 1 see only a single summary route.

Task 6

In Area 1, configure R1 to advertise its loopback interfaces with their correct mask. You should use a minimum number of commands to accomplish this request. Configure the appropriate router such that the summary route (1.1.0.0/22) plus subnet 1.1.2.0/24 is advertised to other areas.

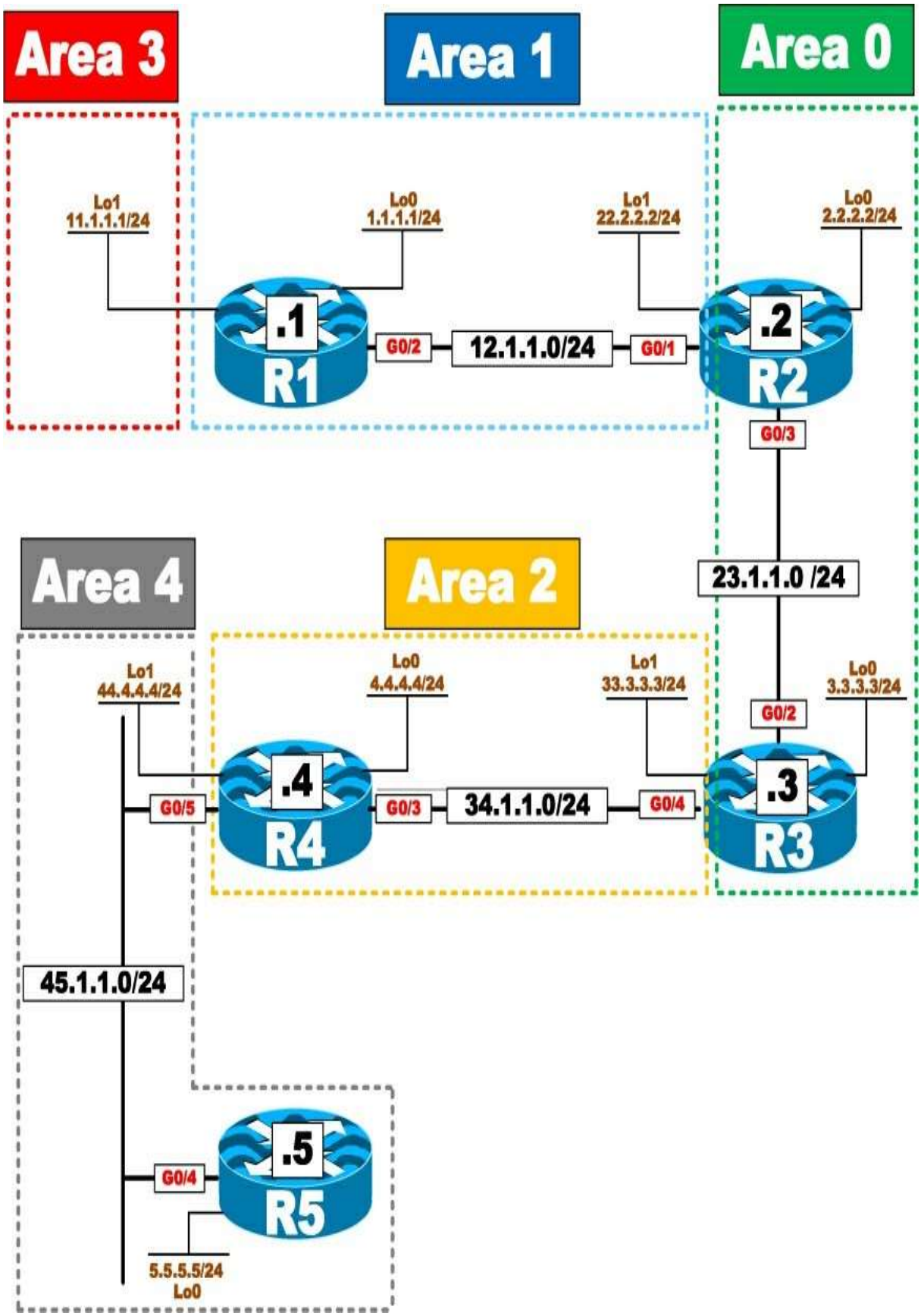
Task 7

In Area 2, configure the appropriate router(s) such that R3 advertises the summary plus one of the specific routes (4.4.0.4/24).

Task 8

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 9: Virtual Links and GRE Tunnels


Area 3
Area 1
Area 0
Lo1
11.1.1.1/24
Lo0
1.1.1.1/24
Lo1
22.2.2.2/24
Lo0
2.2.2.2/24
.1
R1
G0/2
12.1.1.0/24
G0/1
.2
R2
G0/3
23.1.1.0 /24
Area 4
Area 2
Lo1
44.4.4.4/24
Lo0
4.4.4.4/24
Lo1
33.3.3.3/24
Lo0
3.3.3.3/24
G0/2
.4
R4
G0/5
G0/3
34.1.1.0/24
G0/4
.3
R3
45.1.1.0/24
.5
R5
G0/4
5.5.5.5/24
Lo0

This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 9-Virtual-links and GRE Tunnels in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-9.

Task 1

Configure the routers in the previous diagram based on the following chart. The loopback interfaces should be advertised with their correct masks. You may not see all the networks in every router.

Router	Router ID	Interface – Area
R1	0.0.0.1	Lo0 – Area 1 lo1 – Area 3 G0/2 – Area 1
R2	0.0.0.2	Lo0 – Area 0 lo1 – Area 1 G0/1 – Area 1 G0/3 – Area 0
R3	0.0.0.3	Lo0 – Area 0 lo1 – Area 2 G0/2 – Area 0 G0/4 – Area 2
R4	0.0.0.4	Lo0 – Area 2 lo1 – Area 4 G0/3 – Area 2 G0/5 – Area 4
R5	0.0.0.5	Lo0 – Area 4 G0/4 – Area 4

Task 2

Ensure that the networks advertised in Area 3 are reachable by the other routers. *Do not* use a GRE tunnel to accomplish this task.

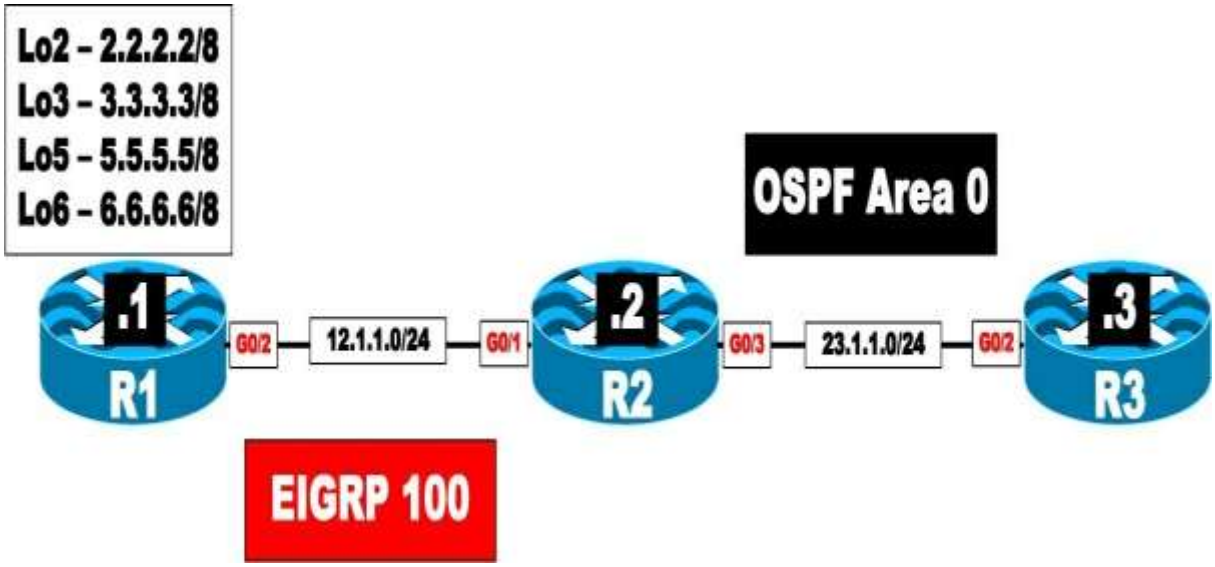
Task 3

Ensure that the networks in Area 4 are reachable by the other routers. You should use a GRE tunnel to accomplish this task. The IP address of the tunnel should be based on the Lo1 interfaces of R3 and R4. *Do not* reconfigure the Loopback1 interfaces of these two routers.

Task 4

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 10: Default Route Injection



This lab should be conducted on the Enterprise POD.

Task 1

Configure the routers in the previous topology. *Do not* configure any routing protocols.

Task 2

Configure EIGRP AS 100 on R1 and all of its directly connected interfaces and R2's G0/1 interface.

Task 3

Configure OSPF on R3's G0/2 interface and R2's G0/3 interfacein Area 0.

Task 4

Configure R2 to inject a default route into the OSPF routing domain if networks 2.0.0.0/8 *or* 3.0.0.0/8 are up.

Task 5

Remove the configurations from the previous task and configure the appropriate router/s based on the following policy:

- R2 should inject a default route into the OSPF routing domain *only* if networks 2.0.0.0/8 and 3.0.0.0/8 are both up.
- *Do not* use route maps, access lists, or prefix lists to complete this task.

Task 6

Remove the configurations from the previous task and configure the appropriate router/s based on the following policy:

- R2 should inject a default route into the OSPF routing domain *only* if network 2.0.0.0/8 is up and network 3.0.0.0/8 is down.

Task 7

Remove the configurations from the previous task and configure the appropriate router/s based on the following policy:

- R2 should inject a default route into the OSPF routing domain *only* if networks 2.0.0.0/8 and 3.0.0.0/8 are both up.
- R2 should *not* configure a static default route.

- R2 should configure a prefix list to accomplish this task.

Task 8

Remove the configurations from the previous task and configure the appropriate router/s, based on the following policy:

- R2 should inject a default route into the OSPF routing domain if: Network 2.0.0.0/8 is up

AND

- Network 3.0.0.0/8 is down

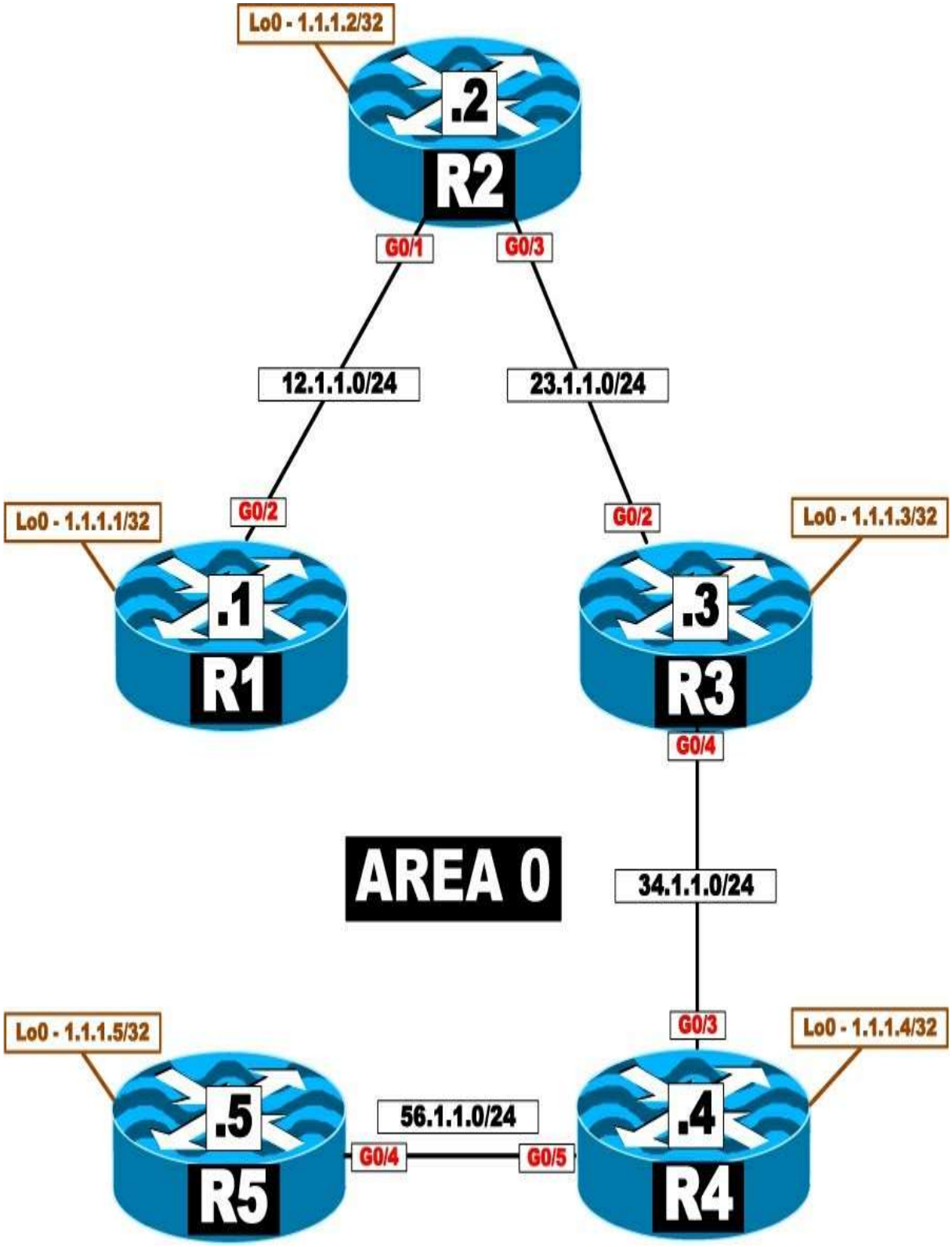
AND

- Networks 5.0.0.0/8 **OR** 6.0.0.0/8 are up

Task 9

Erase the startup configuration and reload the devices before proceeding to the next lab.

Lab 11: OSPF Authentication



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 11-OSPF Authentication in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-11.

Task 1

Configure the directly connected interfaces of all routers in Area 0. The router IDs of the routers in this area should be configured as 0.0.0.*x*, where *x* is the router number.

Task 2

Configure plaintext authentication on all the links connecting the routers in this area. You *must* use a router configuration command as part of the solution in resolving this task. Use **aaa** as the password for this authentication.

Task 3

Remove the authentication configuration from the previous task and ensure that every router sees every route advertised in Area 0.

Task 4

Configure MD5 authentication on all the links in Area 0. You should use a router configuration command as part of the solution to this task. Use **ccc** as the password for this authentication.

Task 5

Remove the authentication configuration from the previous task and ensure that every router sees every route advertised in Area 0.

Task 6

Configure MD5 authentication on the link connecting R1 to R2. You should use a router configuration command as part of the solution to this task. The password should be **ccie**.

Task 7

Reconfigure the authentication password on R1 and R2 to be **CCIE12**, without interrupting the link's operation. *Do not* remove any commands to accomplish this task.

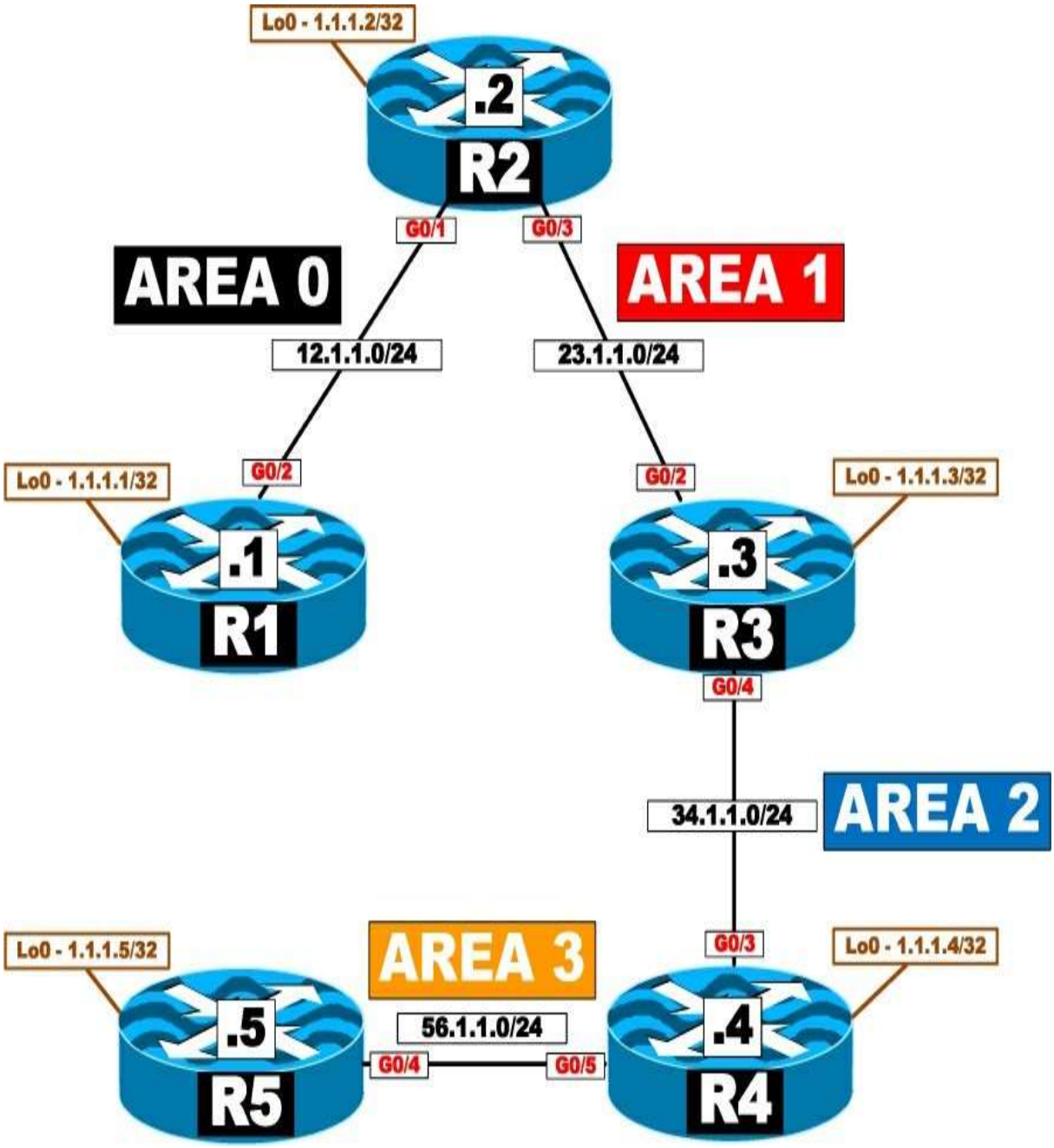
Task 8

Configure MD5 authentication on the link that connects R4 to R5, using **Cisco45** as the password. You should *not* use a router configuration mode to accomplish this task.

Task 9

Reconfigure OSPF areas based on the following chart:

Router	Interface	Area
R1	G0/2	0
	Loopback0	0
R2	G0/1	0
	G0/3	1
	Loopback0	1
R3	G0/2	1
	G0/4	2
	Loopback0	2
R4	G0/3	2
	G0/5	3
	Loopback0	3
R5	G0/4	3
	Loopback0	3



Task 10

Configure MD5 authentication on the link between R1 and R2 in Area 0. The password for this authentication should be set to **Micronics**. You should use router configuration mode to accomplish this task.

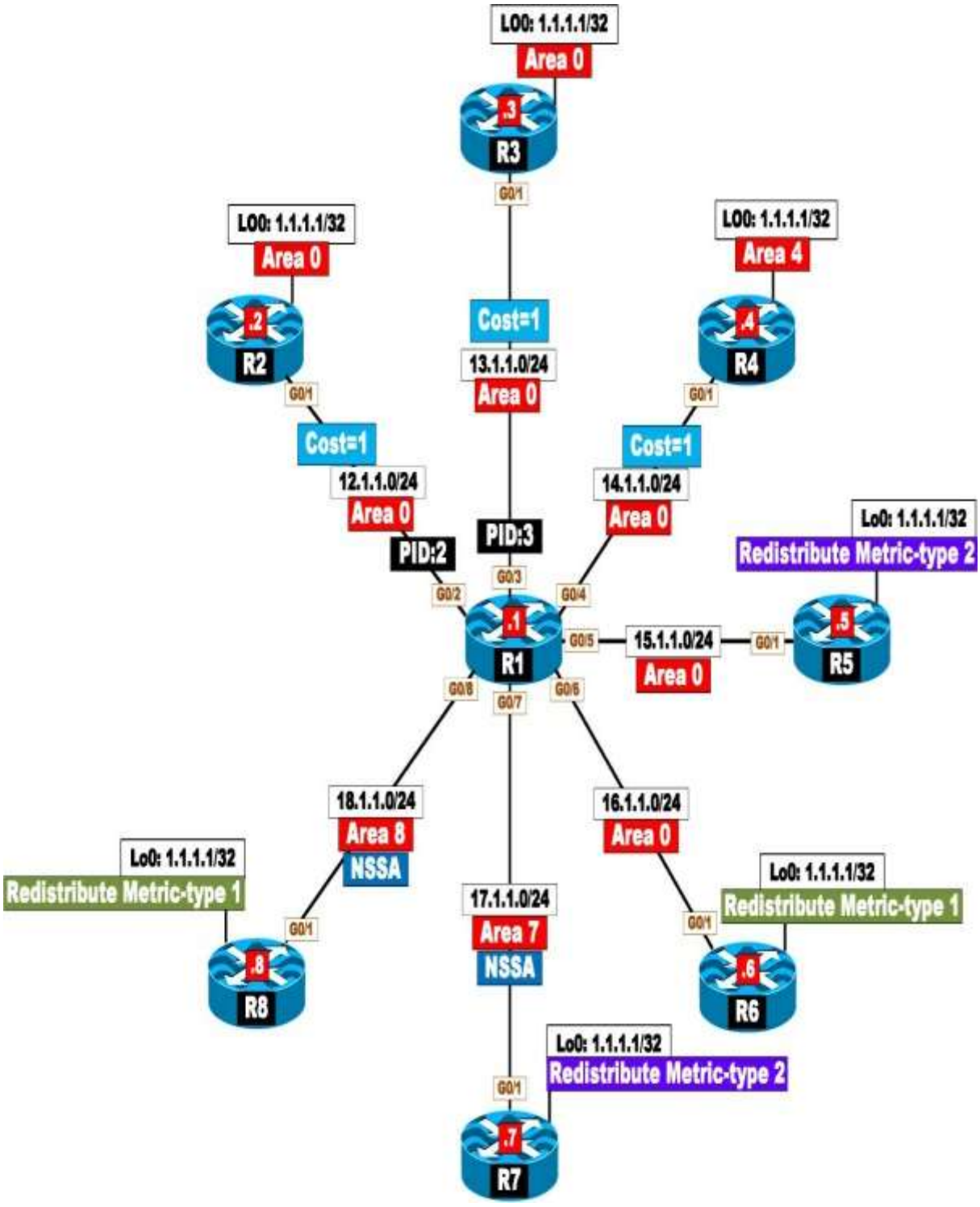
Task 11

Configure the strongest authentication between R4 and R5. Configure the password to be **PSWD**.

Task 12

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 12: OSPF Best-Path Determination



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab 12-OSPF Bestpath Determination in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-12.

Task 1

Configure OSPF on the routers in the previous topology based on the following policy:

- Configure OSPF Area 0 on the link connecting R1 to R2. R2 should runOSPF Area 0 on all of its directly connected interfaces. R1 should use OSPF PID 12, and R2 should use OSPF PID 1.
- Configure OSPF Area 0 on the link connecting R1 to R3. R3 should runOSPF Area 0 on all of its directly connected interfaces. R1 should use OSPF PID 13, and R3 should use OSPF PID 1.
- Configure OSPF Area 0 on the link connecting R1 to R4. R4 should runOSPF Area 0 on its G0/1 interface and Area 4 on its Loopback0 interface.
- Configure OSPF Area 0 on the link connecting R1 to R5. R5 shouldredistribute its Loopback0 interface as external Type 2 in this routing domain.
- Configure OSPF Area 0 on the link connecting R1 to R6. R6 shouldredistribute its Loopback0 interface as external Type 1 in this routing domain.
- Configure OSPF Area 7 on the link connecting R1 to R7. Area 7 shouldbe configured as an NSSA. R7 should redistribute its Loopback0 interface as external Type 2 in this routing domain.

- Configure OSPF Area 8 on the link connecting R1 to R8. Area 8 shouldbe configured as an NSSA. R8 should redistribute its Loopback0 interface as external Type 1 in this routing domain.

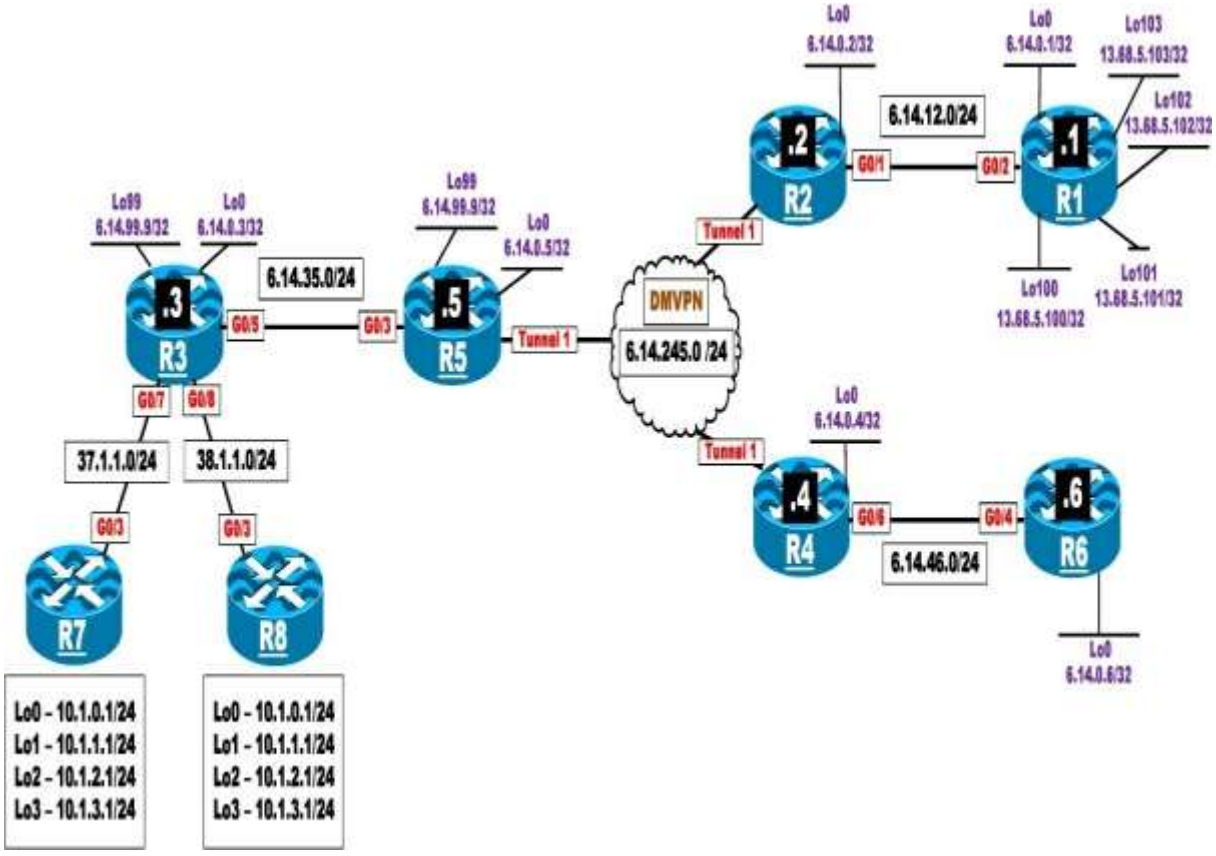
Task 2

A Walk through OSPF best-path determination.

Task 3

Erase the startup configuration and reload the routers before proceeding to the next lab.

Lab 13: OSPF Challenge Lab



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tickets and use Lab 13-OSPF Challenge Lab in the OSPF folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → OSPF folder → Lab-13.

Rules:

- **The DMVPN should *not* be changed to dynamic.**
- **You should *only* fix the problem that is specified in the ticket.**

Ticket 1

R5 can't ping R3's Lo0 interface.

Ticket 2

R2 can't reach R5's Lo0 interface. *There should be a DR*. Do not use **neighbor** command when fixing this problem.

Ticket 3

R1 can't ping R5's Lo0 interface. You *must* use an OSPF command to fix this problem.

Ticket 4

R4 is configured to filter R3's Lo0 interface from reaching R6. R6 should have reachability to all loopback interfaces in this topology except R3's Lo0 interface. However, this is *not* the case. R6 should not get *any* error messages regarding tunnel interfaces. Use two other commands to fix other problems on R6. R6 should not have 6.14.0.3/32 in its routing table after you fix the other problems.

Ticket 5

R5 is configured to inject a default route if R3's Lo0 interface is reachable. However, R5 is *not* injecting a default route. You can remove and reconfigure a command on the appropriate router once, but you *must* use the same method.

Ticket 6

R2 is configured to summarize R1's loopback interfaces (Lo100–Lo103). However, R5 can see all the specific routes of the summary in its routing table. *Do not* remove any commands to accomplish this task.

Ticket 7

R3 is participating in another OSPF routing domain, using the process ID 378. R3 has two neighboring routers, R7 and R8. R7 and R8 are both summarizing their identical specific routes. R3 should use R7 and *not* R8 to reach the summary route. If router R7 is down, then R3 should go through R8 to reach the summary. *Do not* use the following on any router to accomplish this task:

- Cost
- Bandwidth
- Static routes
- PBR
- Any kind of tunneling

- Running extra routing process

Ticket 8

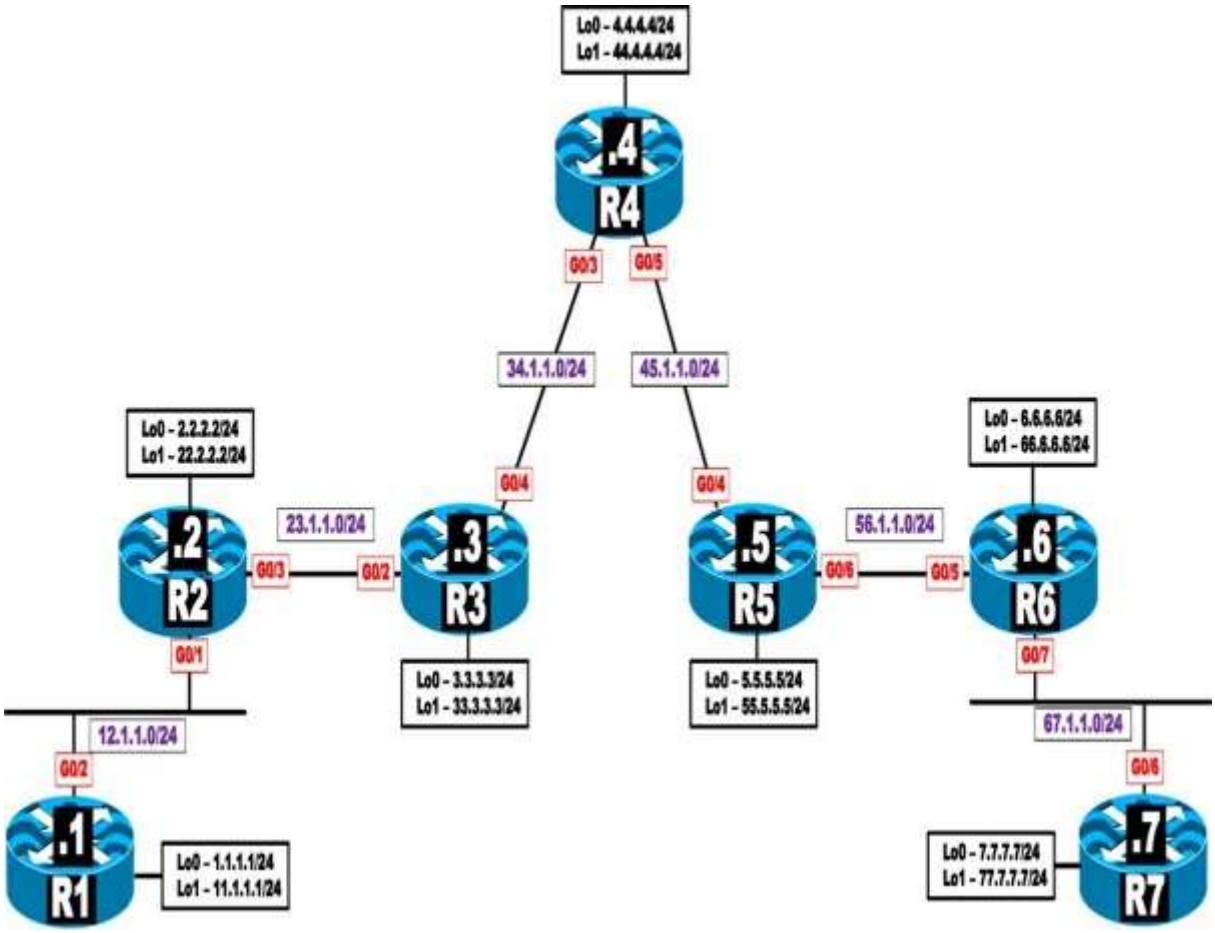
Erase the startup configuration and reload the devices before proceeding to the next lab.

Chapter 7. BGP [This content is currently in development.]

This content is currently in development.

Chapter 8. MPLS and L3VPNs

Lab 1: Configuring Label Distribution Protocol



This lab should be conducted on the Enterprise POD.

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology" folder, ignore the following tasks and use

Lab-1- Configuring Label Distribution Protocol in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-1.

Task 1

Configure OSPF Area 0 on all links in the previous topology except the Loopback1 interfaces. Configure the OSPF router IDs of these routers to be 0.0.0.*x*, where *x* is the router number.

Task 2

Configure Label Distribution Protocol on the links interconnecting the routers in this topology. Ensure that the LDP ID is based on the IP address assigned to the Loopback0 interfaces of these routers. You may override a command from the previous task to accomplish this task.

Task 3

Configure the interval of discovery to be 15 seconds, with a hold timer of 45 seconds on all LSRs.

Task 4

Configure the session keepalives and hold timers of all routers to 30 and 90 seconds, respectively.

Task 5

Configure the LDP router ID of R1 to be its Loopback1 interface. You should not reload the router to accomplish this task.

Task 6

The label space of the routers is platform dependent. By default, the routers begin numbering the labels from 16 up to 100,000. Change the label space such that the routers use the following labels:

Router	Label Range
R1	100–199
R2	200–299
R3	300–399
R4	400–499
R5	500–599
R6	600–699
R7	700–799

Task 7

Examine and describe the control plane for the 7.7.7.7/32 prefix.

Task 8

Examine and describe the data plane for the 7.7.7.7/32 prefix, starting from R1

Task 9

Configure LDP conditional label advertising to exclude the links that interconnect the routers in this topology.

Task 10

To test the effects of TTL propagation, remove the **mpls ip** command from the G0/6 interface of R7, the G0/7 interface of R6, the G0/2 interface of R1, and the G0/1 interface of R2. R1 and R7 will pose as customer routers that do not have MPLS enabled. From R7, you will test the connection to 1.1.1.1 by using a traceroute.

Task 11

Reconfigure the appropriate router/s such that a traceroute from R7 to 1.1.1.1 or from R1 to 7.7.7.7 will not display the links from the provider's network.

Task 12

Remove the **mpls ip** command from all interfaces of the routers within the cloud—that is, R2, R3, R4, R5, and R6. Verify the configuration.

Task 13

Enable LDP on all the links connecting the routers to each other. Do not use the **mpls ip** interface configuration command or a global configuration command to accomplish this task.

Task 14

Configure a GigabitEthernet connection between R3 and R5, using the following parameters and policy:

- R3: G0/5, 35.1.1.3 /24
- R5: G0/3, 35.1.1.5 /24

These links should be included in OSPF Area 0.

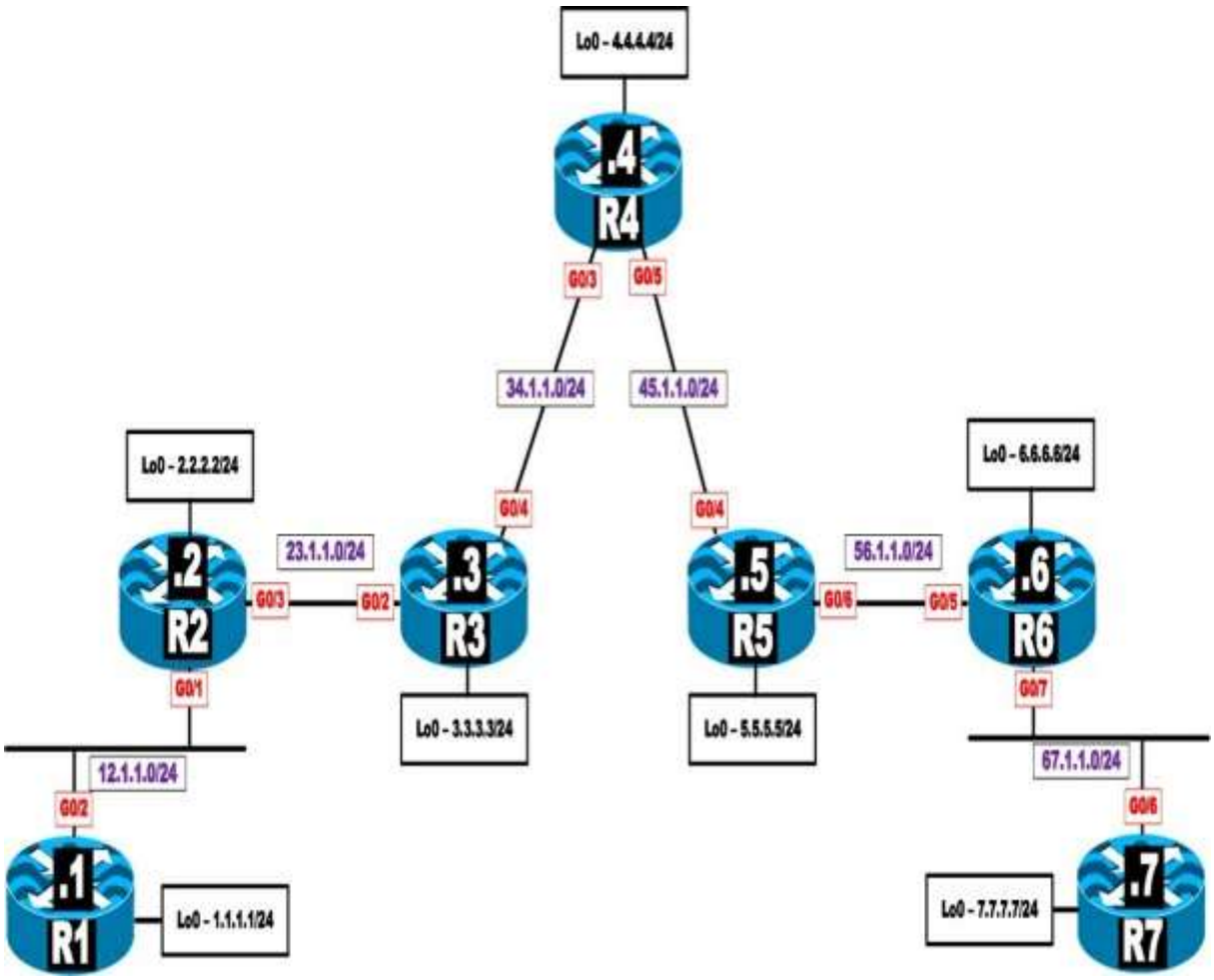
Task 15

Configure the appropriate router/s such that a failure in one of the links between R3 and R5 does not tear down the LDP session between the two LSRs. Do not configure a GRE or an IPnIP tunnel to accomplish this task.

Task 16

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 2: Static and RIPv2 Routing in a VPN



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-2- Static and RIPv2 Routing in a VPN in the MPLS folder in EVENG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-2.

Task 1

Configure OSPF on the core MPLS routers, R2 through R6. Run OSPF Area 0 on the Lo0 interfaces and the links that connect these routers to each other.

Configure the router IDs of R2, R3, R4, R5, and R6 as 0.0.0.2, 0.0.0.3, 0.0.0.4, 0.0.0.5, and 0.0.0.6, respectively.

Task 2

Configure LDP between the core routers (R2 through R6). Ensure that each of these routers uses its Loopback0 interface as its LDP router ID. The core MPLS routers (R2 through R6) should use the following label ranges:

- R2: 200–299
- R3: 300–399
- R4: 400–499
- R5: 500–599
- R6: 600–699

Task 3

Configure MP-iBGP between R2 and R6 as they represent the provider edge routers in this topology in AS 100. Do not allow the BGP peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure virtual routing and forwarding (VRF) instances on R2 and R6 with the following RD and RT values:

- On R2, a VRF instance named CA for Customer A (R1)

- Route distinguisher (RD): 1:10
- Route target (RT): 1:100
- On R6, a VRF instance named CB for Customer A (R2)
- Route distinguisher (RD): 1:20
- Route target (RT): 1:100

Task 5

Configure a static default route on each customer router located in VRF instances CA and CB. Configure these static routes to point to their respective PE router (R2 for R1 and R6 for R7). The PE routers (R2 and R6) should each be configured with a static route that reaches the loopback and the GigabitEthernet interface of the customer router. R2 and R6 should be able to see both static routes in their BGP tables.

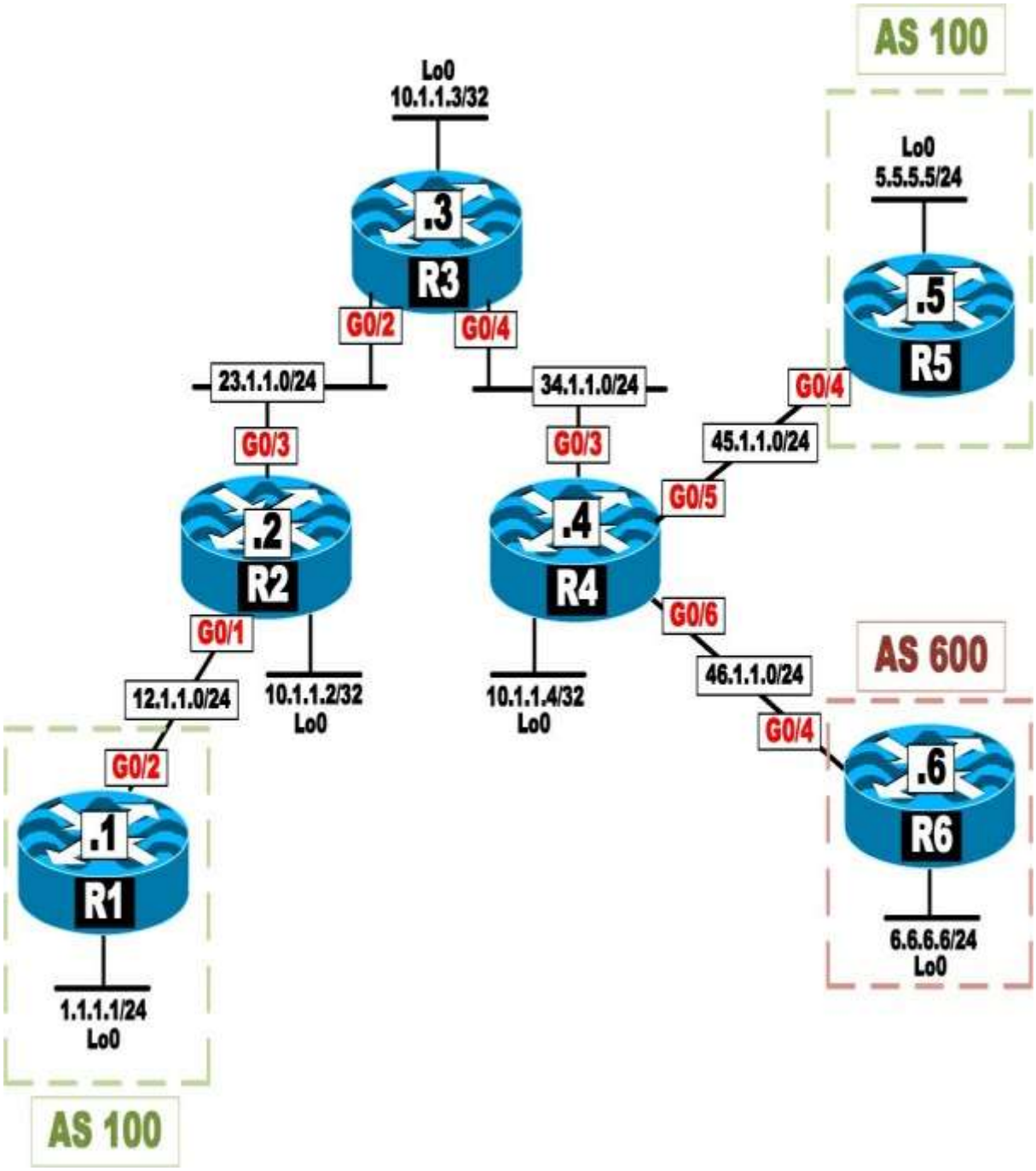
Task 6

Remove the static routes and replace the current method of routing between the PEs and customers with the RIPv2 routing protocol.

Task 7

Erase the startup configuration of these routers and reload the devices before proceeding to the next lab.

Lab 3: EIGRP Routing in a VPN



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-3-EIGRP Routing in a VPN in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-3.

Task 1

Configure OSPF on the core MPLS routers R2(PE-2), R3(P-3), and R4(PE4). Run OSPF Area 0 on the:

- G0/3 interface of R2
- G0/3 interface of R4
- G0/2 and G0/4 interfaces of R3
- Loopback interfaces of these three routers

The router IDs of these routers should be set to 0.0.0.*x*, where *x* is the router number.

Task 2

Configure LDP between the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs. The core MPLS routers (R2, R3, and R7) should use the following label ranges:

- R2: 200–299
- R3: 300–399
- R4: 400–499

Task 3

Configure an MP-BGP peer session for AS 100 between R2 and R4 as they represent the provider edge routers in this topology. Do not allow the BGP

peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure VRF instances on R2 and R4 and enable VRF forwarding on the interfaces of these two routers based on the following chart:

PE	VRF	RD	RT	Interface
R2	11	1:10	1:156	G0/1
R4	55 66	1:50 1:60	1:156 1:156	G0/5 G0/6

You should configure an address family when configuring VRF 66.

Task 5

Configure the following:

1. EIGRP AS 100 between R1 and R2 (PE-2)
2. EIGRP AS 100 between R4 (PE-4) and R5
3. EIGRP 600 between R4 (PE-4) and R6

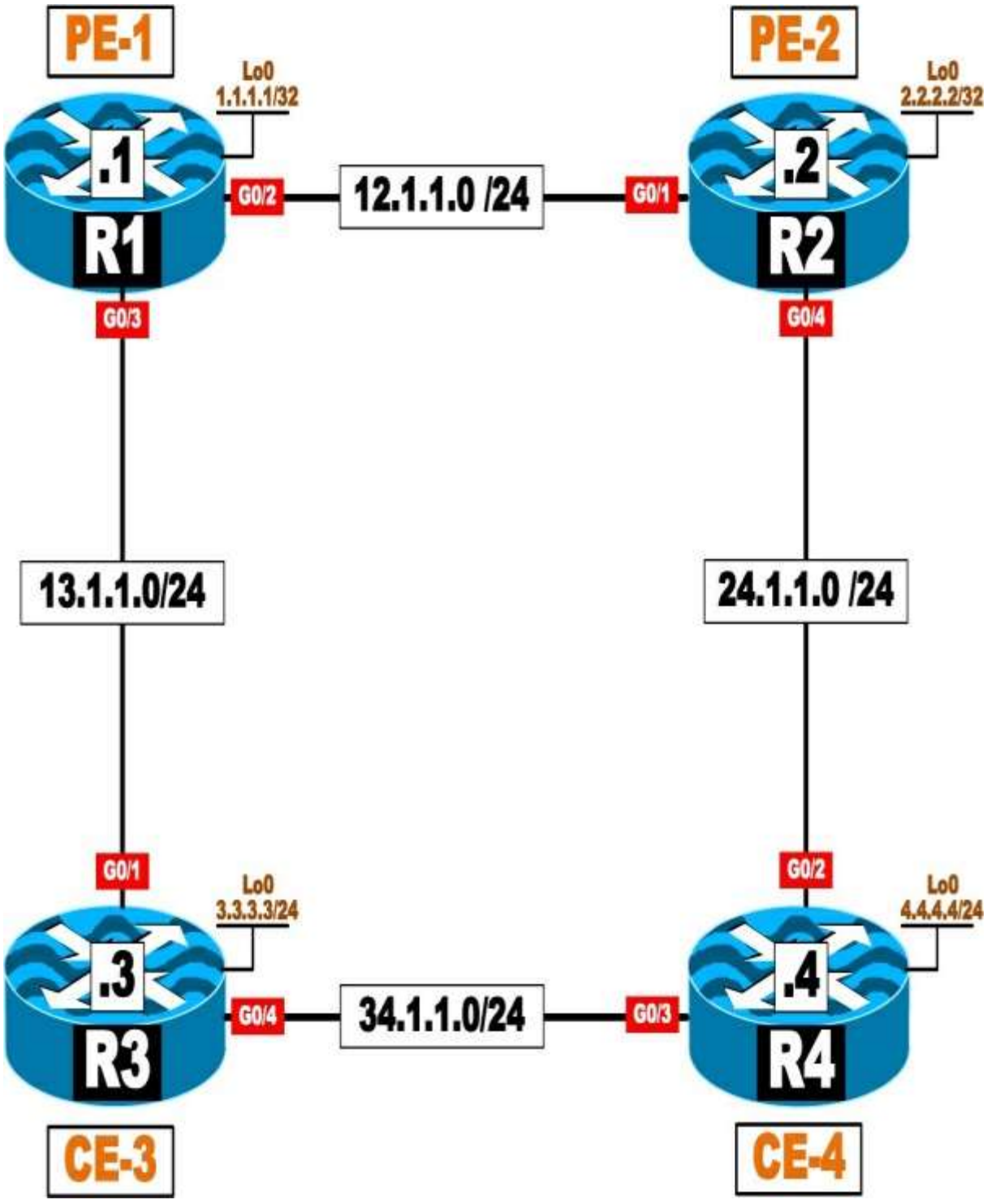
Task 6

Configure the PE routers (R2 and R4) so the CE routers (R1, R5, and R6) can see EIGRP routes advertised from the other CE routers and have reachability to them.

Task 7

Erase the startup configuration of these routers and reload the devices before proceeding to the next lab.

Lab 4: EIGRP Site-of-Origin


PE-1
Lo0
1.1.1.1/32
.1
R1
G0/2
12.1.1.0 /24
G0/1
PE-2
Lo0
2.2.2.2/32
.2
R2
G0/3
G0/4
13.1.1.0/24
24.1.1.0 /24
G0/1
Lo0
3.3.3.3/24
.3
R3
G0/4
34.1.1.0/24
G0/3
G0/2
Lo0
4.4.4.4/24
.4
R4
CE-3
CE-4

Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-4-EIGRP Site-of-Origin in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-4.

Task 1

Configure OSPF Area 0 on the following interfaces:

- R1: Lo0 and G0/2
- R2: Lo0 and G0/1

Configure the router IDs of these two routers as 0.0.0.*x*, where *x* is the router number.

Configure EIGRP AS 100 in named mode on the following interfaces:

- R3: Lo0 and G0/4
- R4: Lo0 and G0/3

Task 2

Configure the PE routers (R1 and R2) to support MPLS VPN using AS 65001 and using their Loopback0 interfaces. Use the following parameters for VRF configuration:

VRF name	TST
RD on R1	1:10
RD on R2	1:20
Route target on both	34:34
PE-CE routing protocol	EIGRP 100

Ensure full connectivity between the customer's (R3 and R4) routes. You should configure named mode where possible.

Task 3

Configure the appropriate routers to prevent the local routes from being learned from the backbone. *Do not* configure R3 or R4, access lists, or prefix lists to accomplish this task.

Task 4

After configuring the previous task, it is obvious that there is no redundancy. If the CE routers (R3 and R4) lose their directly connected link, they will have no reachability to each other's routes. Configure the appropriate router/s based on the following policy:

- If the link between R3 and R4 is up, they should use each other as thenext hop to reach the routes they are advertising.

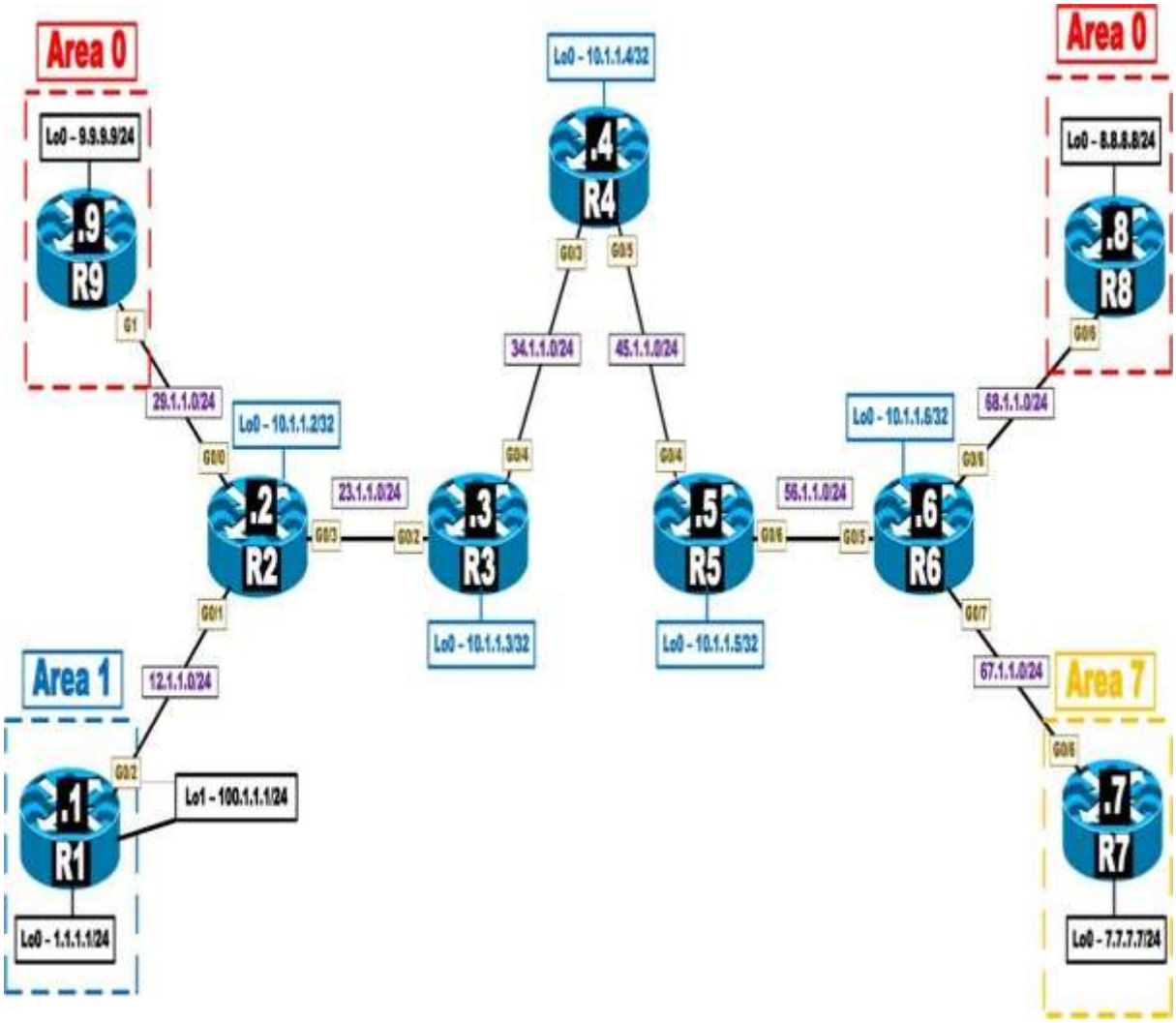
- If the link between R3 and R4 is down, they should go through thecloud to reach each other's routes.

You should configure and test two different solutions to accomplish this task.

Task 5

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 5: OSPF Routing in a VPN



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-5- OSPF Routing in a VPN in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-5.

Task 1

Configure OSPF on the core MPLS routers (R2, R3, R4, R5, and R6). Run OSPF Area 0 on the following:

- G0/3 interface of R2
- G0/2 and G0/4 interfaces of R3
- G0/3 and G0/5 interfaces of R4
- G0/4 and G0/6 interfaces of R5
- G0/5 interface of R6
- Loopback0 interfaces of these routers

Configure the router IDs of these routers to be 0.0.0.*x*, where *x* is the router number.

Task 2

Configure LDP on the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs. The core MPLS routers (R2, R3, R4, R5, and R6) should use the following label ranges:

- R2: 200–299
- R3: 300–399
- R4: 400–499

- R5: 500–599
- R6: 600–699

Task 3

Configure MP-BGP between R2 and R6 as they represent the provider edge routers in this topology in AS 100. Do not allow the BGP peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure VRF instances on R2 and R6 and enable VRF forwarding on the interfaces of these two routers based on the following chart:

Router	VRF	RD	RT	Interface
R2	99 11	1:90 1:10	1:100 1:100	G0/0 G0/1
R6	88 77	1:80 1:70	1:100 1:100	G0/8 G0/7

Task 5

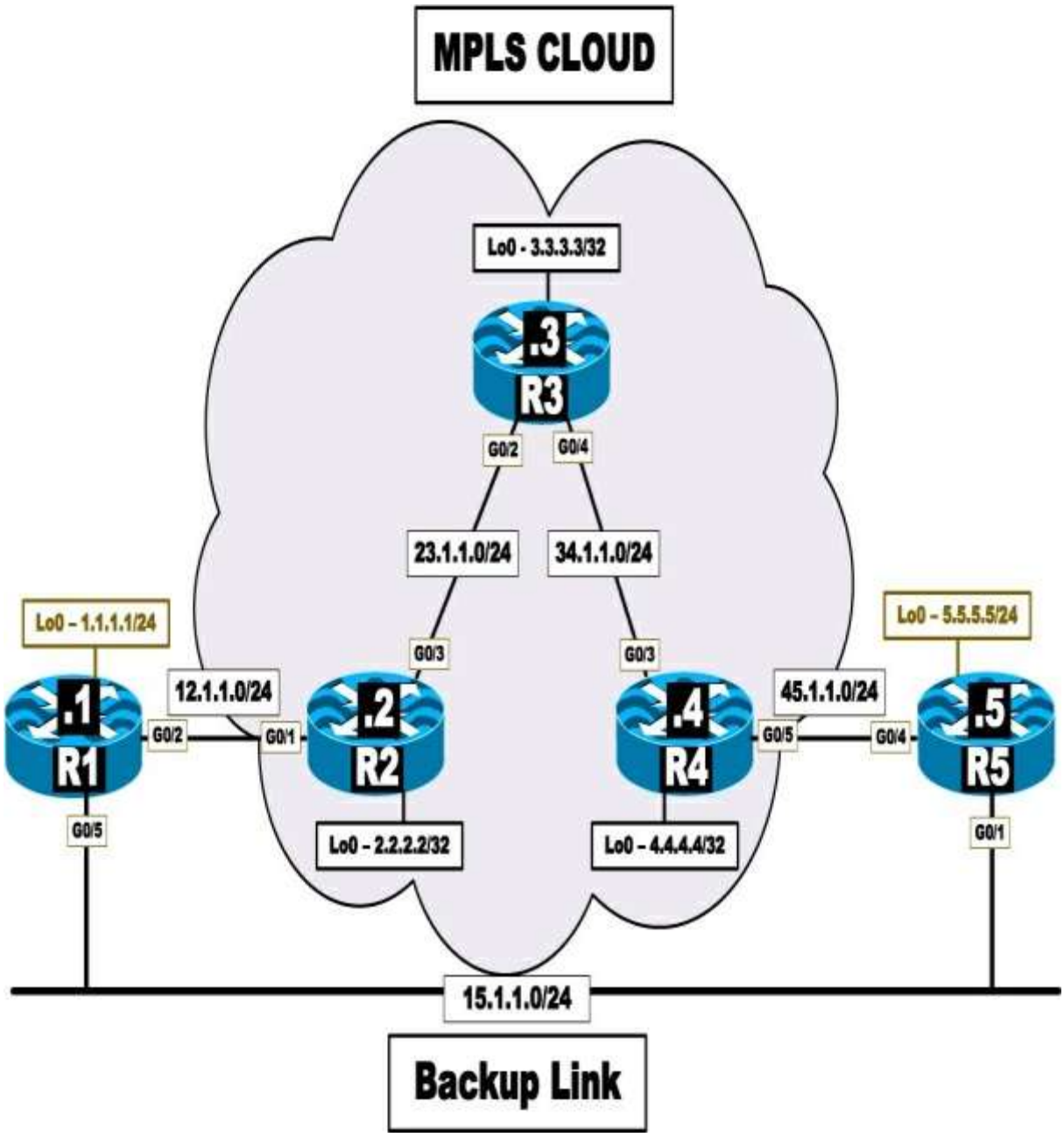
Configure customers R1, R9, R7, and R8 with a VRF service that incorporates OSPF as the routing protocol. R2 should use OSPF process IDs 11 and 99 for R1 and R9, respectively. R6 should use process IDs 77 and 88 for R7 and R8, respectively.

- All directly connected interfaces of R1 should be configured in Area 1except R1's Lo1 interface. R1 should redistribute its Lo1 interface in this routing domain.
- All directly connected interfaces of R9 should be configured in Area 0.
- All directly connected interfaces of R7 should be configured in Area 7.
- All directly connected interfaces of R8 should be configured in Area 8.

Task 6

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 6: Backdoor Links and OSPF



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-6-Backdoor Links and OSPF in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-6.

Task 1

Configure OSPF on the core MPLS routers (R2, R3, and R4). Run OSPF Area 0 on the links and Lo0 interfaces interconnecting these routers. Configure the router IDs of these routers to be 0.0.0.*x*, where *x* is the router number.

Task 2

Configure LDP between the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs. The core MPLS routers (R2, R3, and R4) should use the following label ranges:

- R2: 200–299
- R3: 300–399
- R4: 400–499

Task 3

Configure MP-BGP between R2 and R4 as they represent the provider edge routers in this topology in AS 100. Do not allow the BGP peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure a virtual routing and forwarding (VRF) instance with the name aa, the route distinguisher (RD) 1:10, and the route target (RT) 1:100 for R1 (the customer) on R2 (the PE router). On R4, use the same route targets for the VRF instance but configure the RD to be 1:50 and the name of the VRF to be aa.

Task 5

Configure OSPF using the same process ID on the customer routers (R1 and R5). Configure MP-iBGP such that the customer routers can see each other's routes. Customer routers should advertise their Lo0 interfaces with the correct mask.

Task 6

Configure the G0/5 interface of the customer router R1 and the G0/1 interface of the customer router R5 as a backup link. Ensure that the MPLS service is preferred over the backup link.

Task 7

Provide another solution to the problem that is different from the one in the previous task.

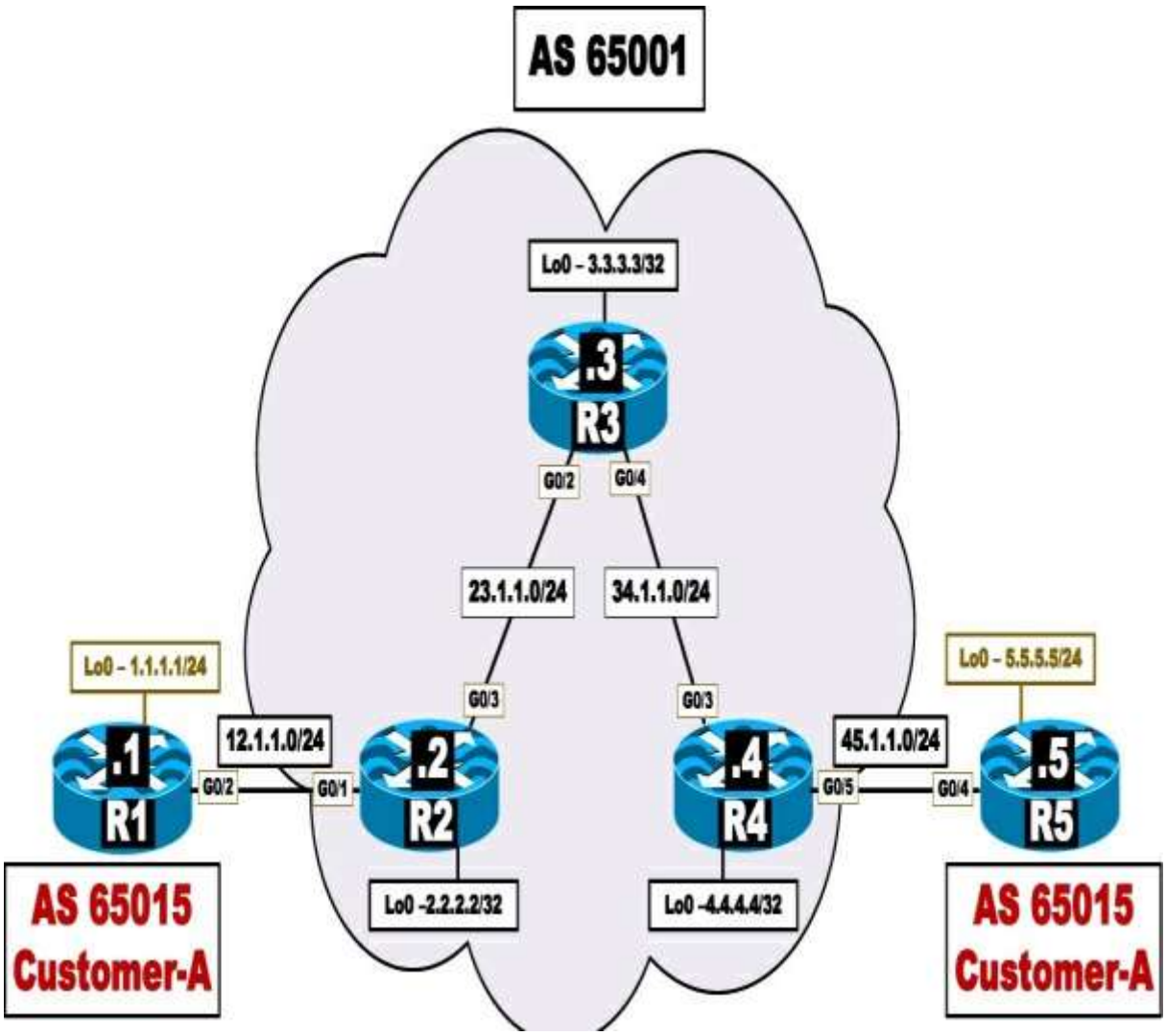
Task 8

To satisfy the customer, who is complaining about two additional routes in their routing table, remove them. These are the IP addresses assigned to the Lo24 interface of R2 and Lo42 interface of R4.

Task 9

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 7: BGP Routing in a VPN



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-7- BGP Routing in a VPN in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-7.

Task 1

Configure OSPF on the core MPLS routers (R2, R3, and R4). Run OSPF Area 0 on the links and Lo0 interfaces interconnecting these routers. Configure the router IDs of these routers to be 0.0.0.*x*, where *x* is the router number.

Task 2

Configure LDP between the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs. The core MPLS routers (R2, R3, and R4) should use the following label ranges:

- R2: 200–299
- R3: 300–399
- R4: 400–499

Task 3

Configure MP-BGP between R2 and R4 as they represent the provider edge routers in this topology in AS 100. Do not allow the BGP peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure a virtual routing and forwarding (VRF) instance with the name aa, the route distinguisher (RD) 1:10, and the route target (RT) 1:100 for R1 (the customer) on R2 (the PE router). On R4, use the same route targets for the VRF instance but configure the RD to be 1:50 and the name of the VRF to be aa.

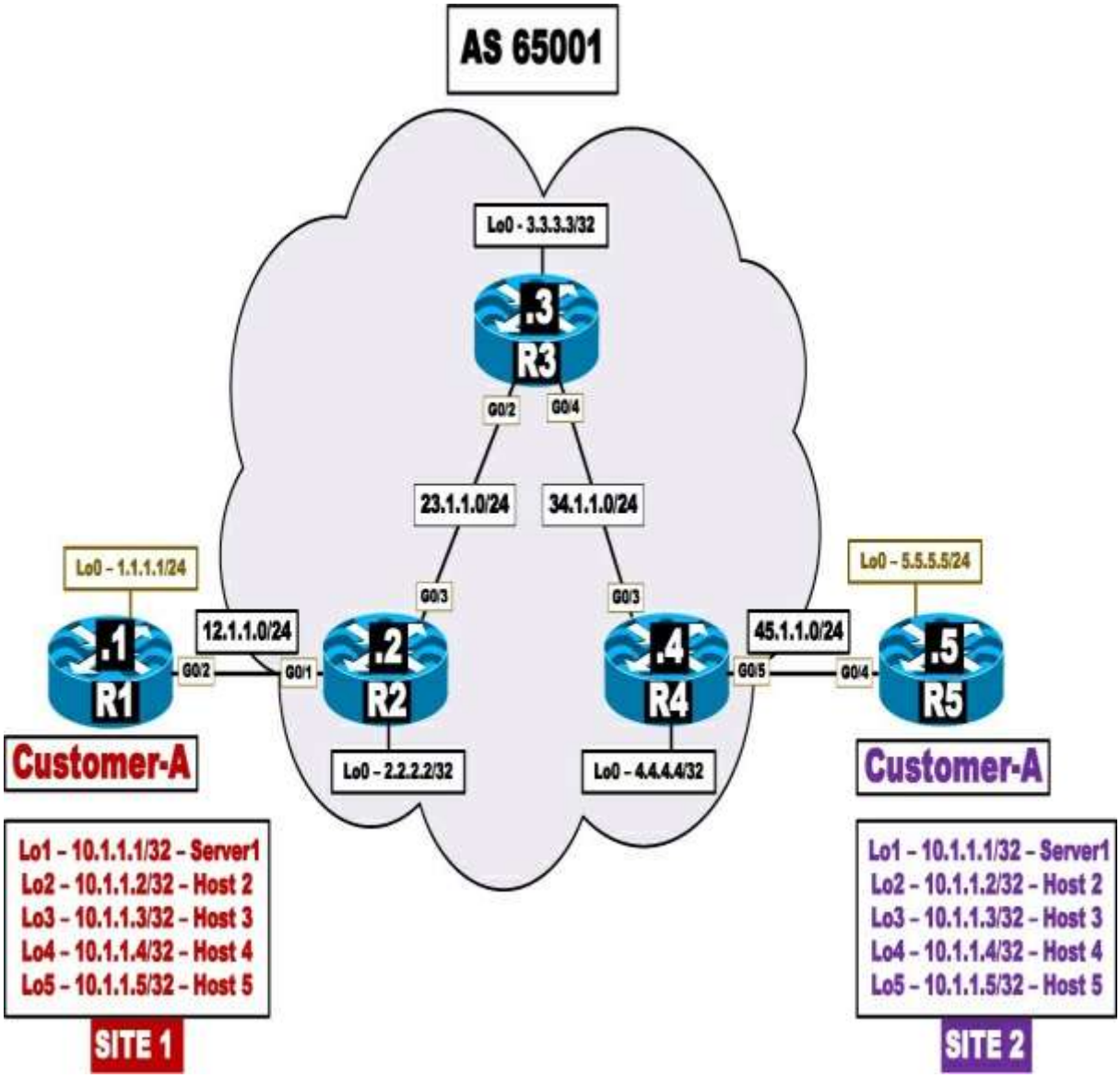
Task 5

Configure BGP as the MPLS routing context between the CEs (R1 and R5) and their respective PEs (R2 and R4). The customer AS of 65015 should be assigned to both customer sites.

Task 6

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 8: MPLS and NAT



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-8- MPLS and NAT in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-8.

Task 1

Configure OSPF on the core MPLS routers (R2, R3, and R4). Run OSPF Area 0 on the links and Lo0 interfaces interconnecting these routers. Configure the router IDs of these routers to be 0.0.0.*x*, where *x* is the router number. Configure the CE routers, R1 and R5, with a static default route pointing to their next hop router.

Task 2

Configure LDP between the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs.

Task 3

Configure MP-BGP between R2 and R4 as they represent the provider edge routers in this topology in AS 100. The only BGP peering relationship should be VPNv4. These two neighbors should use their Lo0 interfaces for their peering.

Task 4

Configure the following VRF instances, RDs, and route targets on the PE routers, based on the following chart:

Router	VRF Name	RD	Route Target	Interface
R2	111	1:10	route-target both 1:100	G0/1
R4	555	1:50	route-target both 1:100	G0/5

Task 5

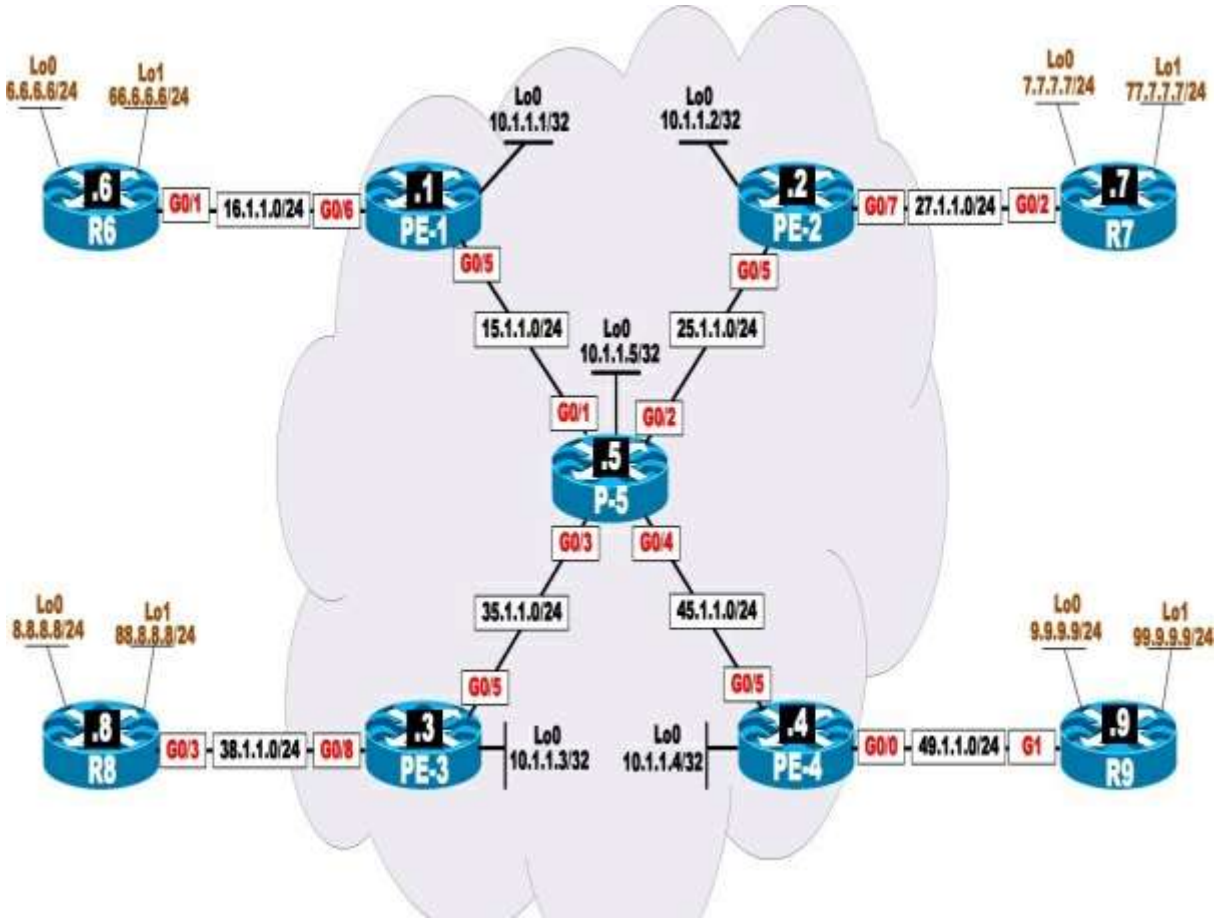
Ensure that the hosts in Site-1 can access the server in Site-2 and vice versa. Configure NAT on the CE routers (R1 and R5). Use the following translation chart:

Router	Inside Local	Inside Global
R1	10.1.1.1 10.1.1.2–10.1.1.5	100.1.1.1 100.1.1.2–100.1.1.5
R5	10.1.1.1 10.1.1.2–10.1.1.5	200.1.1.1 200.1.1.2–200.1.1.5

Task 6

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 9: Route Targets, Import Maps, and Export Maps



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-9-Route-Targets-Import maps and Export maps in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-9.

Task 1

Configure OSPF on the core MPLS routers (PE-1, PE-2, PE-3, PE-4, and P5). Run OSPF Area 0 on the G0/5 interfaces of PE-1(R1), PE-2(R2), PE-3 (R3), and PE-4(R4), the G0/1, G0/2, G0/3, and G0/4 interfaces of P-5(R5), and the loopback interfaces of these routers. Configure the router IDs of these routers to be 0.0.0.*x*, where *x* is the router number.

Task 2

Configure LDP between the core routers. These routers should use their Loopback0 interfaces as their LDP router IDs. The core MPLS routers (PE-1, PE-2, PE-3, PE-4, and P-5) should use the following label ranges:

1. PE-1: 100–199
2. PE-2: 200–299
3. PE-3: 300–399
4. PE-4: 400–499
5. P-5: 500–599

Task 3

Configure MP-BGP peer sessions between all PE routers using AS 100. Do not allow the BGP peers to share IPv4 routing information by default. The only BGP peering relationship should be VPNv4.

Task 4

Configure VRF instances and RDs on the PE routers and enable VRF forwarding on the interfaces of these routers based on the following chart:

Router	VRF	RD	Interface
PE-1	66	1:60	G0/6
PE-2	77	1:70	G0/7
PE-3	88	1:80	G0/8
PE-4	99	1:90	G0/0

Do not configure route targets.

Task 5

Configure routing between the CE and the PE routers, based on the following chart:

CE Router	PE Router	Routing Protocol
R6	PE-1	EIGRP 100
R7	PE-2	OSPF area 0
R8	PE-3	RIPv2
R9	PE-4	BGP AS 200

Task 6

Configure the appropriate PE/s such that routers R6 and R7 can exchange routes and be in the same VPN.

Task 7

Configure the appropriate PE/s such that routers R7 and R9 can exchange routes and be in the same VPN.

Task 8

Configure the appropriate PE/s such that routers R8 and R9 can exchange routes and be in the same VPN.

Task 9

Configure the appropriate PE/s such that routers R6 and R8 can exchange routes and be in the same VPN.

Task 10

Configure PE-1 such that R6 only receives networks 7.7.7.7/32 and 8.8.8.0/24. This may affect the reachability achieved in some of the previous tasks.

Task 11

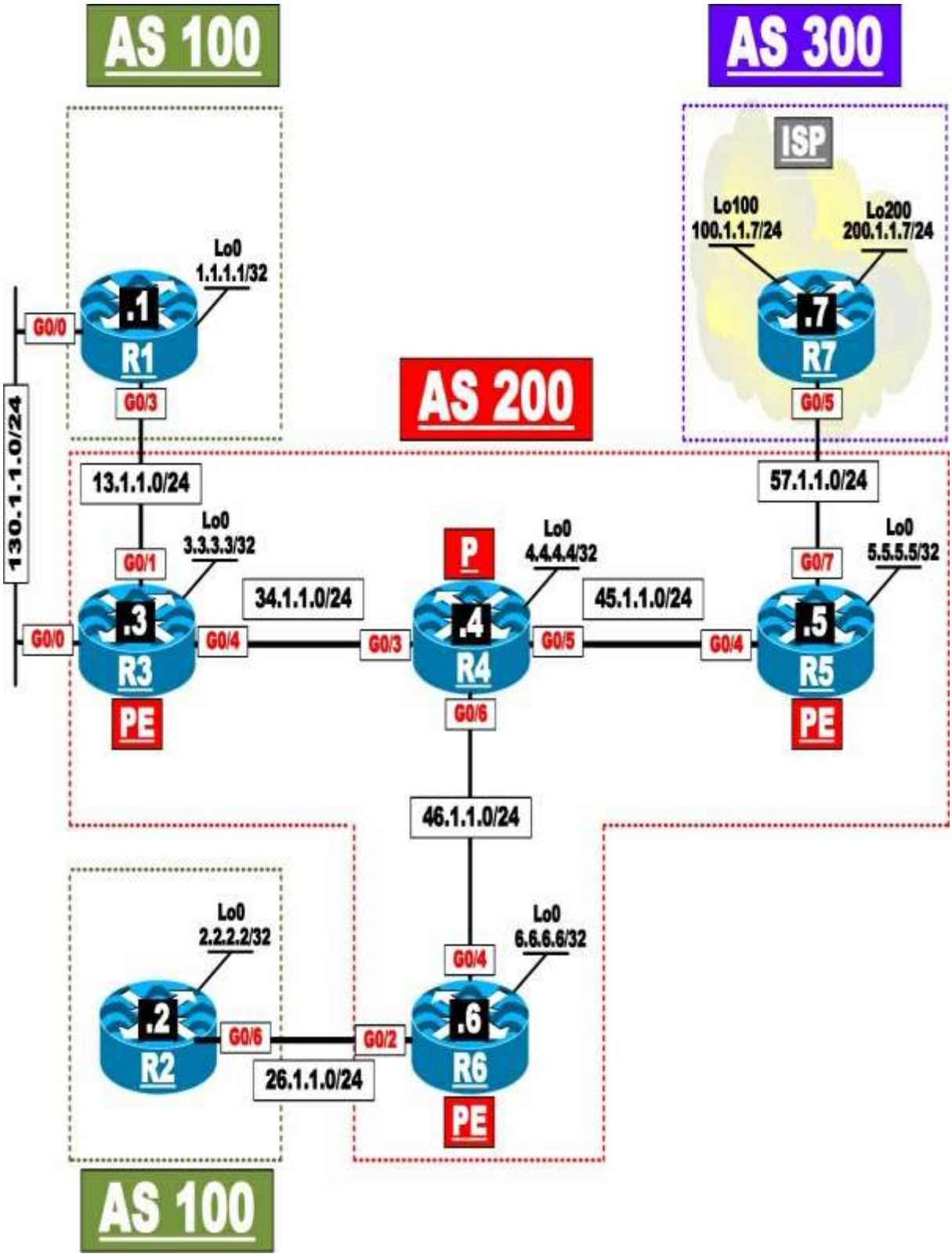
Configure the following loopback interfaces on R6, R7, and R8:

- R6: Loopback200, 200.1.1.6/32
- R7: Loopback200, 200.1.1.7/32
- R8: Loopback200, 200.1.1.8/32

Task 12

Erase the startup configuration of the routers and reload the devices before proceeding to the next lab.

Lab 10: Internet Access Methods: Partial Internet Routes



Lab Setup:

If you are using EVE-NG, and you have imported the EVE-NG topology from the EVE-NG-Topology folder, ignore the following tasks and use Lab-10-Internet Access Methods Partial Internet Routes in the MPLS folder in EVE-NG.

To copy and paste the initial configurations, go to the Initial-config folder → MPLS folder → Lab-10.

Task 1

Configure the core routers (R3, R4, R5, and R6) to support MPLS VPN using AS 200.

- R5 in AS 200 should be configured with an IPv4 peering to the physicalinterface of R7 in AS 300. R7 should advertise its Lo100 and Lo200 interfaces in this AS.
- R3 in AS 200 should be configured with an IPv4 peering to the G0/0interface of R1 in AS 100.

Task 2

Configure a VRF instance called aaa on R3 and R6 (the PE routers).

- Apply this VRF instance to the G0/1 interface of R3 facing R1 and theG0/2 interface of R6 facing R2 (the CE routers).
- Use the RD 1:10 and the route target 1:100 on R3 and the RD 1:20 withthe same RT configured on R3.

Task 3

Ensure the customers R1 and R2 only receive partial internet routes using the following policies:

- *Only* R1 should receive the partial Internet routes.
- R2 should go through R1 to reach the partial routes.

Task 4

Erase the startup configuration on all routers and reload the devices before proceeding to the next lab.

Chapter 9. IPv6 [This content is currently in development.]

This content is currently in development.

Chapter 10. SD-WAN [This content is currently in development.]

This content is currently in development.

Chapter 11. SD-ACCESS [This content is currently in development.]

This content is currently in development.