

JNCIP

Juniper® Networks
Certified Internet Professional
Study Guide - Chapter 4

by Harry Reynolds

This book was originally developed by Juniper Networks Inc. in conjunction with Sybex Inc. It is being offered in electronic format because the original book (ISBN: 0-7821-4073-4) is now out of print. Every effort has been made to remove the original publisher's name and references to the original bound book and its accompanying CD. The original paper book may still be available in used book stores or by contacting, John Wiley & Sons, Publishers. www.wiley.com.

Copyright © 2003-6 by Juniper Networks Inc. All rights reserved.

This publication may be used in assisting students to prepare for a Juniper JNCIP exam but Juniper Networks Inc. cannot warrant that use of this publication will ensure passing the relevant exam.





Chapter

4

IS-IS Configuration and Testing

JNCIP LAB SKILLS COVERED IN THIS CHAPTER:

- ✓ **Multi-Level IS-IS configuration**
 - Default route origination
- ✓ **Network types**
- ✓ **Authentication**
 - Hello and LSP authentication
- ✓ **IS-IS policy**
 - Summarization, filtering, and route leaking
 - Route Redistribution
- ✓ **Metrics, timers, and various other “knobs”**



This chapter details various JNCIP-level IS-IS configuration scenarios, and provides examples of the verification methods that can be used to confirm proper IS-IS protocol operation. It is assumed that your configurations are currently based on the case study criteria presented at the end of the initial system and interface configuration chapters. You will now be adding the IS-IS protocol to this test bed. If you are unsure as to the state of your routers, you should compare your configuration against those provided at the end of Chapters 1 and 2, and verify that all of your router interfaces are operational before proceeding.

As described in the previous chapter, the proper operation of your IGP will play a critical role in the overall success of your JNCIP lab attempt. Serious problems with your IGP's operation will almost guarantee that you will experience malfunctions with higher-level protocols and services like BGP or RSVP signaling, and the cascading effects of the cumulative point loss that will occur in such situations will virtually assure that you will experience little joy at the end of your testing day. A successful JNCIP exam candidate must possess the practical skills and protocol knowledge necessary to proficiently configure and troubleshoot the IS-IS protocol in a variety of network scenarios as demonstrated in this chapter.

Because the overall purpose of OSPF and IS-IS are essentially identical, this chapter will demonstrate IS-IS configuration and testing using topologies and configuration requirements that parallel the examples used in the previous chapter wherever possible. The fact that IS-IS supports only the point-to-point and broadcast network types means that you will be mercifully spared the need to configure multipoint and NBMA network types.

This chapter concludes with a case study that is designed to approximate a typical JNCIP IS-IS configuration scenario. The results of key operational mode commands are provided in the case study analysis section so that you can also compare the behavior of your network to a known good example. Router configurations that meet all case study requirements are provided at the end of the case study for comparison with your own configurations.

Multi-Level IS-IS Configuration

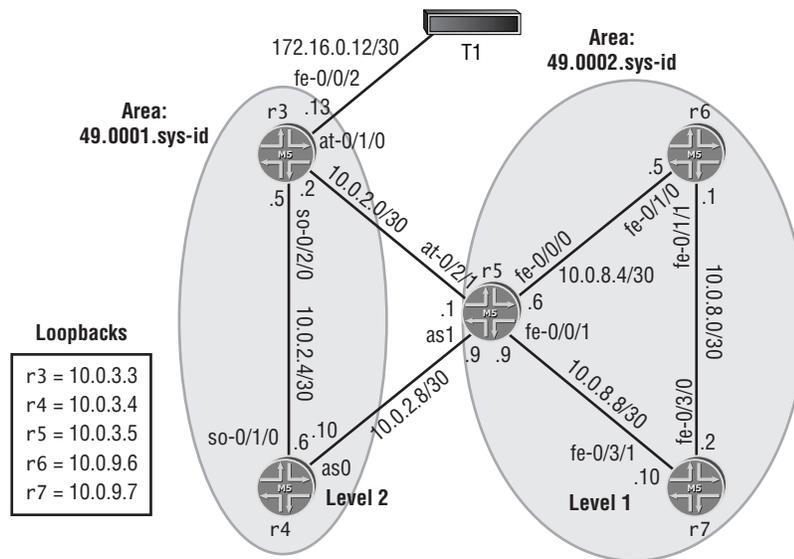
Like OSPF, the IS-IS protocol supports the partitioning of a routing domain into multiple levels. The use of multiple levels will generally improve protocol scalability, as level 2 (backbone) Link State Protocol Data Units (LSPs) are normally not flooded into a level 1 area.

Briefly stated, an IS-IS level 2 area is analogous to the OSPF backbone area (0), while a level 1 area operates much as an OSPF Totally Stubby Area, in that a default route is normally used to reach both inter-level and AS external routes. Unlike OSPF, IS-IS area boundaries occur between

routers, such that a given Intermediate System (IS) is always wholly contained within a particular area. Level 1 adjacencies can only be formed between routers that share a common area number, while a level 2 adjacency can be formed between routers that may or may not share an area number. Juniper Networks M-series routers support the IS-IS protocol strictly for use in building IP routing tables. There is no support for the routing of OSI Connectionless Network Service (CLNS) network layer packets.

We begin our IS-IS configuration example with the network topology shown in Figure 4.1.

FIGURE 4.1 Multi-level IS-IS



Referring to Figure 4.1, you can see that r5 will need to function as a level 1/level 2 router to interconnect the level 2 backbone (area 49.0001) and the level 1 area (49.0002) containing r6 and r7. To complete this configuration example, your IS-IS configuration must meet the following criteria:

- The System ID (SysID) must be based on the router's IPv4 lo0 address.
- Loss of any individual interface must not totally disrupt the router's IS-IS operation.
- The IPv4 lo0 addresses of all routers must be reachable via IS-IS.
- The link between r3 and T1 must appear in area 49.0001 as an intra-area route. Ensure that no IS-IS adjacencies can be established on this interface.
- The loopback addresses of level 2 routers may not appear in a level 1 area.
- Only one adjacency per router pairing is permitted.

IS-IS Configuration

We begin by configuring *r4* as a level 2 IS-IS router based on the topology shown earlier in Figure 4.1. To enable IS-IS routing on a Juniper Networks router, you must add the `iso` protocol family to each IS-IS interface, assign your router's IS-IS address, and enable the IS-IS routing instance. What follows are step-by-step instructions on how to configure IS-IS and confirm its operation.

Configure Interface Support of IS-IS PDUs

The following commands are required to add the `iso` protocol family to all of *r4*'s interfaces that are expected to run IS-IS. Failing to include the `iso` family will cause IS-IS to be disabled on that interface, because such an interface will not be able to send or receive link-level IS-IS PDUs:

```
[edit interfaces]
lab@r4# set so-0/1/0 unit 100 family iso
```

```
[edit interfaces]
lab@r4# set as0 unit 0 family iso
```

The IS-IS protocol will not operate until you have assigned a NET to one of your router's interfaces. The term *NET* is a fancy word for the IS-IS address that will uniquely identify a given router within an IS-IS routing domain. You must assign the NET to the router's `lo0` interface to meet this example's requirement that "loss of any individual interface must not totally disrupt the router's IS-IS operation." You must also ensure that the NET is in some way traceable to the IPv4 address assigned to the router's `lo0` address. The following command correctly configures *r4*'s NET for this example:

```
[edit interfaces lo0 unit 0 family iso]
lab@r4# set address 49.0001.0100.0000.3004.00
```

Though perhaps not intuitive, the IS-IS address (NET) that has been assigned to *r4* is interpreted as follows:

- The Authority and Format Identifier (AFI) value of `49` indicates a private use/locally assigned ISO NET.
- The area number is `0001`.
- The 6-byte System ID (SysID) is set to encode the router's IPv4 `lo0` address in Binary Coded Decimal (BCD) format using three BCD digits for each decimal-delimited portion of corresponding IPv4 address. The resulting SysID is interpreted as `10.0.3.4`, which meets the configuration requirements of this example. Although numerous mapping approaches could be used to satisfy the requirements of this example, e.g., `0010.0000.0034`, it is highly recommended that you adopt a consistent transcoding algorithm to ensure ease of SysID interpretation, and to help prevent the assignment of a duplicate SysID.

- The Selector field is set to *00*, which makes this an ISO NET and not a Network Service Access Point Address (NSAPA). You should always specify an all-zeros selector byte for proper IS-IS operation.

The IS-IS related interface configuration of *r4* is displayed in Listing 4.1. The highlights call out the recently added IS-IS configuration:

Listing 4.1: r4 IS-IS Interface Configuration

```
[edit]
lab@r4# show interfaces as0
aggregated-sonet-options {
    minimum-links 2;
    link-speed oc3;
}
unit 0 {
    family inet {
        address 10.0.2.10/30;
    }
    family iso;
}
[edit]
lab@r4# show interfaces so-0/1/0
dce;
hold-time up 30 down 30;
encapsulation frame-relay;
lmi {
    n392dce 2;
    n393dce 3;
    t392dce 25;
    lmi-type itu;
}
unit 100 {
    dlci 100;
    family inet {
        address 10.0.2.6/30;
    }
    family iso;
}
[edit]
lab@r4# show interfaces lo0
unit 0 {
    family inet {
```

```

        address 10.0.3.4/32;
    }
    family iso {
        address 49.0001.0100.0000.3004.00;
    }
}
[edit]
lab@r4# show interfaces fxp0
unit 0 {
    family inet {
        address 10.0.1.4/24;
    }
}

```



To help guard against inadvertent use of IS-IS on the OoB management network, you should avoid adding the `iso` family to the router's `fxp0` interface. If you are using configuration groups to expedite the addition of protocol families, you must use caution to ensure either that the configuration group does not “catch” the `fxp0` interface or that you explicitly disable IS-IS on the `fxp0` interface when making use of the `interface all` option shown in this chapter. IGP adjacency formation over the OoB network can be disastrous because your routers will be inclined to forward traffic over the `fxp0` subnet (because of the lower metric associated with this path), in conjunction with the fact that transit traffic is not supported over the `fxp0` interface. Can you say “black hole”?

Configure *r4*'s IS-IS Routing Instance

The following commands are used to enable the main IS-IS instance and to correctly associate *r4*'s IS-IS capable interfaces with IS-IS level 2 operation. Unlike OSPF, you must explicitly list the router's `lo0` interface under the `[edit protocols isis]` stanza, as this interface is the source of the router's NET, and therefore must be configured as an IS-IS interface. The `lo0` interface will operate in the passive mode by default, which is ideal because adjacency formation can never occur on a virtual interface anyway.

```

[edit protocols isis]
lab@r4# set interface lo0 level 1 disable

[edit protocols isis]
lab@r4# set interface as0 level 1 disable

[edit protocols isis]
lab@r4# set interface so-0/1/0.100 level 1 disable

```

By disabling level 1 IS-IS operation on r4's area 49.0001 interfaces, you ensure that at most one adjacency (level 2) can be formed between each router pair in area 49.0001. The JUNOS software default is to enable interface operation at both levels 1 and 2, which would result in two adjacencies forming, and subsequent point loss on your exam because the criteria for this example limits you to one adjacency per router pairing. The resulting configuration of r4's main IS-IS routing instance is shown here:

```
[edit protocols isis]
lab@r4# show
interface so-0/1/0.0 {
    level 1 disable;
}
interface as0.0 {
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}
```

Verify IS-IS on r4

After committing the initial IS-IS configuration, we can display the state of r4's IS-IS interfaces to confirm their proper configuration and level assignment. The results indicate that r4's as0, so-0/1/0.100, and lo0 interfaces are enabled for IS-IS operation at level 2 only. As expected, no Designated Intermediate System (DIS) has been elected because of the lack of broadcast interface types on r4:

```
lab@r4> show isis interface
```

```
IS-IS interface database:
```

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
as0.0	2	0x1	<u>Disabled</u>	Point to Point	10/10
lo0.0	0	0x1	<u>Disabled</u>	Passive	0/0
so-0/1/0.100	2	0x1	<u>Disabled</u>	Point to Point	10/10

```
lab@r4>
```

In the absence of other IS-IS routers, we expect to find that r4 has no neighbors and, therefore, no adjacencies:

```
lab@r4> show isis adjacency
```

```
lab@r4>
```

As predicted, r4 currently has no IS-IS adjacencies. We can obtain additional information about the default IS-IS timers and source of the router's NET by using the `detail` switch, as shown next:

```
lab@r4> show isis interface detail
IS-IS interface database:
as0.0
  Index: 10, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: disabled
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    2           0       64    10      9      27
lo0.0
  Index: 3, State: 0x7, Circuit id: 0x1, Circuit type: 0
  LSP interval: 100 ms, CSNP interval: disabled, Sysid: r4
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    2           0       64     0 Passive
so-0/1/0.100
  Index: 11, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: disabled
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    2           0       64    10      9      27
```

This display confirms that the `lo0` interface is the source of the router's NET, that no adjacencies have been formed, and that the default (non-Designated Intermediate System [DIS]) hello and hold timers are being used on all interfaces except `lo0`, which does not use them because of its passive mode of operation.

IS-IS Timers

Unlike OSPF, IS-IS will permit adjacency formation between systems that have been configured to use different hello timers. The hold interval is used to inform the remote system how long the adjacency can be considered valid in the absence of received hello LSPs. By default, the hold value is three times the configured hello interval, but both parameters can be set independently. The default ratio means that an adjacency can be maintained even if two out of three hello packets are lost. JUNOS software will automatically reduce an interface's hold and hello intervals by $\frac{2}{3}$ rds (3-second hellos and 9-second hold time) should the router be elected the DIS for the network that attaches to that interface. This enables the rapid detection of DIS failures. The IS-IS hold times at each level are unrelated to each other, so you may configure an interface's level 1 and level 2 timers differently with no ill effects.

The last step in confirming IS-IS operation on r4 is to examine the LSDB:

```
lab@r4# run show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
r4.00-00              0x8    0xa284    1195 L1 L2
  1 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
r4.00-00              0xc    0xbab3    1195 L1 L2
  1 LSPs
```

Many exam candidates are surprised to find their router contains a level 1 database entry when they know that all of the router interfaces have been set to run level 2 only. The lone level 1 LSP is the result of disabling level 1 operation on the router's interfaces but not in the IS-IS routing instance itself. This LSP can never leave your router, and no level 1 LSPs can be received from other routers in the routing domain until IS-IS level 1 operation is enabled on at least one interface. A detailed examination of the LSDB confirms that the level 1 LSP is not advertising any prefixes (as yet, no interfaces are enabled for level 1 operation), and also confirms that the level 2 LSP is advertising the route to all three of r4's IS-IS subnetworks:

```
[edit]
lab@r4# run show isis database detail
IS-IS level 1 link-state database:

r4.00-00 Sequence: 0x8, Checksum: 0xa284, Lifetime: 818 secs

IS-IS level 2 link-state database:

r4.00-00 Sequence: 0xd, Checksum: 0xb8b4, Lifetime: 1176 secs
  IP prefix:                10.0.2.4/30 Metric:      10 Internal
  IP prefix:                10.0.2.8/30 Metric:      10 Internal
  IP prefix:                10.0.3.4/32 Metric:       0 Internal
```

If the presence of this LSP serves as a distraction, you can disable IS-IS level 1 in the routing instance by issuing a **set protocols isis level 1 disable** command. This command is not used here because r4 is expected to require level 1 capabilities at a later point in this chapter's examples.

Configure *r5* for IS-IS Operation

Though not shown here, *r5*'s IS-IS configuration begins with the addition of the `iso` family to its `as1`, `at-0/2/1.35`, `fe-0/0/0`, `fe-0/0/1`, and `lo0` interfaces using the commands that were shown for *r4* in the section “Configure Interface Support of IS-IS PDUs” earlier in this chapter. The following commands define the correct interface and area associations needed by *r5*, which functions as a L1/L2 router in this example:

```
[edit protocols isis]
lab@r5# set interface lo0 level 1 disable

[edit protocols isis]
lab@r5# set interface at-0/2/1.35 level 1 disable

[edit protocols isis]
lab@r5# set interface as1 level 1 disable

[edit protocols isis]
lab@r5# set interface fe-0/0/0 level 2 disable

[edit protocols isis]
lab@r5# set interface fe-0/0/1 level 2 disable
```

The resulting multi-level configuration for *r5* is shown next:

```
[edit protocols isis]
lab@r5# show
interface fe-0/0/0.0 {
    level 2 disable;
}
interface fe-0/0/1.0 {
    level 2 disable;
}
interface at-0/2/1.35 {
    level 1 disable;
}
interface as1.0 {
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}
```

An alternative, but equally functional, configuration for `r5` makes good use of the `interface all` keyword to save some typing:

```
[edit protocols isis]
lab@r5# show
interface fe-0/0/0.0 {
    level 2 disable;
}
interface fe-0/0/1.0 {
    level 2 disable;
}
interface all {
    level 1 disable;
}
```

Many operators feel that this alternative configuration “contradicts itself” in that intuition tells them that the `interface all level 1 disable` statement should affect all interfaces, and thereby cause problems with `r5`’s Fast Ethernet interfaces, which are supposed to operate at level 1. This author likes to view the apparent quandary from a “longest match” perspective, in that the router will always override less specific configurations, such as `interface all`, with more specific information such as `interface fe-0/0/0 level 2 disable`. In this case, the `fe-0/0/0` statement is more specific than the `all` keyword, so the router does the right thing by enabling level 1 only on the Fast Ethernet interfaces and level 2 only on all other interfaces. You should pay extra attention to the `show isis interfaces` command results when using this approach. You may find the router is running IS-IS on an unintended interface, such as `fxp0`, due to the non-specific nature of the `all` keyword.

Verify IS-IS on `r5`

After committing `r5`’s IS-IS configuration, we expect to see an adjacency to `r4`, and the correct interface-to-area associations as demonstrated next:

```
[edit]
lab@r5# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
as1.0              2  0x1 Disabled            Point to Point    10/10
at-0/2/1.35        2  0x1 Disabled            Point to Point    10/10
fe-0/0/0.0         1  0x2 r5.02             Disabled          10/10
fe-0/0/1.0         1  0x3 r5.03             Disabled          10/10
lo0.0              0  0x1 Disabled            Passive           0/0
```

These results confirm that `r5`’s interfaces have been correctly configured with the `iso` family, and that the interfaces have been placed into the correct levels as stipulated by Figure 4.1, shown

earlier. You can also see that r5 has elected itself as the DIS on its two broadcast-capable IS-IS interfaces. We now confirm the r4-to-r5 IS-IS adjacency:

```
[edit]
lab@r5# run show isis adjacency detail
r4
  Interface: as1.0, Level: 2, State: Up, Expires in 23 secs
  Priority: 0, Up/Down transitions: 1, Last transition: 00:11:13 ago
  Circuit type: 2, Speaks: IP, IPv6
  IP addresses: 10.0.2.10
```

These results confirm that r5 has a single level 2 adjacency to r4, whose IP address is 10.0.2.10. Because r5 is a L1/L2 attached router, we can now examine the level 1 link-state database associated with area 49.0002 to confirm that loopback addresses from backbone routers are not being advertised into the level 1 area. The contents of r5's LSDB are shown in Listing 4.2.

Listing 4.2: The Contents of r5's IS-IS LSDB

```
[edit]
lab@r5# run show isis database detail
IS-IS level 1 link-state database:

r5.00-00 Sequence: 0x5, Checksum: 0x62c, Lifetime: 1123 secs
  IP prefix:                10.0.8.8/30 Metric:    10 Internal
  IP prefix:                10.0.8.4/30 Metric:    10 Internal

IS-IS level 2 link-state database:

r4.00-00 Sequence: 0x71, Checksum: 0xa9d4, Lifetime: 566 secs
  IS neighbor:              r5.00 Metric:    10
  IP prefix:                10.0.2.4/30 Metric:    10 Internal
  IP prefix:                10.0.2.8/30 Metric:    10 Internal
  IP prefix:                10.0.3.4/32 Metric:    0 Internal

r5.00-00 Sequence: 0x6, Checksum: 0x10f1, Lifetime: 1123 secs
  IS neighbor:              r4.00 Metric:    10
  IP prefix:                10.0.2.8/30 Metric:    10 Internal
  IP prefix:                10.0.2.0/30 Metric:    10 Internal
  IP prefix:                10.0.8.8/30 Metric:    10 Internal
  IP prefix:                10.0.8.4/30 Metric:    10 Internal
  IP prefix:                10.0.3.5/32 Metric:    0 Internal 0x2
0x831 28
```

This display indicates that r5's loopback interface has been correctly configured to run level 2 only. Had level 1 operation been enabled on lo0, r5 would have then included its loopback address in its level 1 LSP, thus violating the configuration requirements given for this example. You can also see that r5 has two level 2 LSPs, one of which was received from its adjacent neighbor, r4. It should be noted that like an OSPF Totally Stubby Area, no backbone (level 2) or external prefixes are leaked into a level 1 area by default. Level 1 prefixes are leaked up into the IS-IS backbone, however, as can be seen by the 10.0.8.x/30 prefixes in r5's level 2 LSP. Based on these results, we find that both r4 and r5 have been correctly configured for this example.

Configure r3 for IS-IS Operation

The following commands are entered on r3 to complete the configuration of the level 2 backbone area. Once again, you may assume that r3's interfaces have been configured with the `iso` family and the correct NET, as demonstrated previously for r4:

```
[edit protocols isis]
lab@r3# set interface at-0/1/0.35 level 1 disable

[edit protocols isis]
lab@r3# set interface lo0 level 1 disable

[edit protocols isis]
lab@r3# set interface so-0/2/0.100 level 1 disable

[edit protocols isis]
lab@r3# set interface fe-0/0/2 passive

[edit protocols isis]
lab@r3# set interface fe-0/0/2 level 1 disable
```

The resulting IS-IS configuration for r3 is shown next:

```
[edit protocols isis]
lab@r3# show
interface fe-0/0/2.0 {
    passive;
    level 1 disable;
}
interface at-0/1/0.35 {
    level 1 disable;
}
```

```
interface so-0/2/0.100 {
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}
```

By setting r3's fe-0/0/2 interface to passive, you ensure that no adjacencies can be formed over the interface and that the route to 172.16.0.12/30 will be injected into area 49.0001 as an internal route. Failing to set this interface as passive would result in loss of points on the JNCIP exam because there is no other way to correctly advertise the 172.16.0.12/30 route as an IS-IS internal prefix while also preventing adjacency formation. It is a good idea to disable level 1 operation on r3's fe-0/0/2 interface, even though it's passive, because this interface is associated with a level 2-only router and such configuration will keep r3's database tidy.

Verify IS-IS on r3

With the IS-IS backbone configured, you will expect to see that r3 has formed a single level 2 adjacency to each of its neighbors, and that the link to T1 is correctly reported in its level 2 LSP. You start by confirming the correct interface and IS-IS level configuration:

```
lab@r3> show isis interface
```

```
IS-IS interface database:
```

Interface	L	CirID	Level 1	DR	Level 2	DR	L1/L2	Metric
at-0/1/0.35	2	0x1	Disabled		Point to Point		10/10	
lo0.0	0	0x1	Disabled		Passive		0/0	
so-0/2/0.100	2	0x1	Disabled		Point to Point		10/10	

The results leave something to be desired in that r3's fe-0/0/2 interface is not listed as an IS-IS interface. This could be the result of the interface being down, the lack of IS-IS instance configuration, or the lack of the iso family on the correct logical unit. A quick look at the interface's status confirms that the lack of family iso on the device's logical unit 0 is the cause of the problem:

```
[edit]
```

```
lab@r3# run show interfaces terse fe-0/0/2
```

Interface	Admin	Link	Proto	Local	Remote
fe-0/0/2	up	up			
fe-0/0/2.0	up	up	inet	172.16.0.13/30	

The following command adds the iso family to the correct logical unit on interface fe-0/0/2:

```
[edit interfaces]
```

```
lab@r4# set fe-0/0/2 unit 0 family iso
```

The results are now confirmed:

```
[edit]
lab@r3# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
at-0/1/0.35        2  0x1 Disabled          Point to Point   10/10
fe-0/0/2.0         0  0x2 Disabled          Passive          10/10
lo0.0              0  0x1 Disabled          Passive          0/0
so-0/2/0.100      2  0x1 Disabled          Point to Point   10/10
```

You now confirm the correct IS-IS adjacency status for area 49.0001:

```
lab@r3> show isis adjacency
Interface          System          L State          Hold (secs) SNPA
at-0/1/0.35        r5              2 Up            22
so-0/2/0.100      r4              2 Up            25
```

Good, r3 has the required level 2 adjacencies to both r4 and r5 as required. Next, confirm the presence of the 172.16.0.12/30 route as an IS-IS level 2 route. We start by confirming the route's presence in r3's level 2 LSP:

```
[edit]
lab@r3# run show isis database r3 detail
IS-IS level 1 link-state database:

r3.00-00 Sequence: 0x12, Checksum: 0x53cd, Lifetime: 789 secs

IS-IS level 2 link-state database:

r3.00-00 Sequence: 0x15, Checksum: 0xb78a, Lifetime: 1052 secs
IS neighbor:                r4.00 Metric:      10
IS neighbor:                r5.00 Metric:      10
IP prefix:                  172.16.0.12/30 Metric: 10 Internal
IP prefix:                  10.0.2.4/30 Metric: 10 Internal
IP prefix:                  10.0.2.0/30 Metric: 10 Internal
IP prefix:                  10.0.3.3/32 Metric: 0 Internal
```

As required, the 172.16.0.12/30 prefix is being reported as an internal level 2 route. The last confirmation is to verify that other backbone routers have correctly installed this route. The following command is issued on r5:

```
[edit]
lab@r5# run show route protocol isis 172/8
```

```
inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
172.16.0.12/30 * [IS-IS/18] 14:19:29, metric 20, tag 2
> to 10.0.2.2 via at-0/2/1.35
```

As expected, r5 has installed the route as an IS-IS level 2 internal route as indicated by the tag 2 value (which denotes level 2) and the preference setting of 18.

Area 49.0002 Configuration

The following commands correctly configure r6 for operation as an IS-IS level 1 router in area 49.0002. You begin by configuring its interfaces with the `iso` family and the router's NET:

```
[edit interfaces]
lab@r6# set fe-0/1/0 unit 0 family iso

[edit interfaces]
lab@r6# set fe-0/1/1 unit 0 family iso

[edit interfaces lo0 unit 0]
lab@r6# set family iso address 49.0002.0100.0000.9006.00
```

The IS-IS routing instance for r6 is now configured. Because r6 is strictly a level 1 router, we disable all level 2 processing and save some typing by using **interface all**, as shown:

```
[edit protocols isis]
lab@r6# set level 2 disable

[edit protocols isis]
lab@r6# set interface all
```

The IS-IS configuration of r6 is shown below:

```
[edit protocols isis]
lab@r6# show
level 2 disable;
interface all;
```

The configuration of r7 is not shown here because it is virtually identical to that of r6 except for interface-naming specifics.

Area 49.0002 Verification

After committing the level 1 IS-IS configuration on both r6 and r7, we expect to find that a level 1 adjacency has formed between them:

```
lab@r6> show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
fe-0/1/1.0	r7	<u>1</u> Up	8	0:a0:c9:6f:7b:1a

With the level 1 adjacency confirmed up, we next verify proper DIS election on the broadcast interface between r6 and r7:

```
lab@r6> show isis interface
```

IS-IS interface database:

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
fe-0/1/0.0	1	0x2	r6.02	Disabled	10/10
<u>fe-0/1/1.0</u>	1	0x3	<u>r7.02</u>	Disabled	10/10
lo0.0	0	0x1	Passive	Disabled	0/0

The display indicates that r7 has won the DIS election on the 10.0.8.0/30 subnet. We confirm this by verifying that r7 has generated a pseudonode LSP for the broadcast network with r6 listed as the only neighbor:

```
lab@r6> show isis database r7 detail
```

IS-IS level 1 link-state database:

```
r7.00-00 Sequence: 0x3, Checksum: 0x7247, Lifetime: 539 secs
  IS neighbor:                               r7.02 Metric:      10
  IP prefix:                                10.0.9.7/32 Metric:    0 Internal
  IP prefix:                                10.0.8.8/30 Metric:   10 Internal
  IP prefix:                                10.0.8.0/30 Metric:   10 Internal
```

```
r7.02-00 Sequence: 0x1, Checksum: 0xfdbc, Lifetime: 539 secs
```

```
  IS neighbor:                               r6.00 Metric:      0
  IS neighbor:                               r7.00 Metric:      0
```

IS-IS level 2 link-state database:

This display confirms that r7 is acting as the pseudonode for the 10.0.8.0/30 subnet, as indicated by its generation of an LSP with a non-zero selector field. Because all level 2 IS-IS processing has been disabled in r6 and r7, we find that the level 2 database is empty as expected.

Verify Overall IS-IS Operation

The spot checks performed thus far have returned the expected results for each of the two IS-IS areas in the configuration example outlined earlier in Figure 4.1, but the astute reader will have noticed that the command outputs displayed in the previous section indicate that r5 has not yet formed the required level 1 adjacencies to the routers in area 49.0002. The lack of an adjacency to the L1/L2-attached r5 results in the absence of the default route needed by level 1 routers to reach other areas:

```
lab@r7> show route protocol isis

inet.0: 11 destinations, 11 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.8.4/30      *[IS-IS/15] 00:26:23, metric 20, tag 1
                 > to 10.0.8.1 via fe-0/3/0.0
10.0.9.6/32     *[IS-IS/15] 00:26:23, metric 10, tag 1
                 > to 10.0.8.1 via fe-0/3/0.0

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

Troubleshooting Adjacency Problems

The lack of an IS-IS adjacency between r5 and area 49.0002 routers could be the result of physical layer problems. This possibility is quickly eliminated through a quick round of ping testing, however:

```
lab@r6> ping 10.0.8.6
PING 10.0.8.6 (10.0.8.6): 56 data bytes
64 bytes from 10.0.8.6: icmp_seq=0 ttl=255 time=0.529 ms
64 bytes from 10.0.8.6: icmp_seq=1 ttl=255 time=0.451 ms
^C
--- 10.0.8.6 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.451/0.490/0.529/0.039 ms
```

These results indicate that physical layer connectivity between r6 and r5 is not the cause of the adjacency problem. Next, reconfirm that r5 has the necessary level 1 configuration:

```
lab@r5> show isis interface
IS-IS interface database:
```

Interface	L CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
as1.0	2	0x1 Disabled	Point to Point	10/10
at-0/2/1.35	2	0x1 Disabled	Point to Point	10/10
<u>fe-0/0/0.0</u>	<u>1</u>	<u>0x2 r5.02</u>	Disabled	10/10
<u>fe-0/0/1.0</u>	<u>1</u>	<u>0x3 r5.03</u>	Disabled	10/10
lo0.0	0	0x1 Disabled	Passive	0/0

The fact that both r5 and r6 have elected themselves as the DIS for the 10.0.8.4/30 subnet provides an indication that we are experiencing some type of “ships in the night” issue that results in each router ignoring the other’s level 1 IS-IS packets.

As with the OSPF protocol, IS-IS tracing, with special attention placed on hello packet exchanges (and any error messages that may be occurring), is one of the best ways to troubleshoot adjacency formation problems. The following traceoptions configuration has been added to r6.

```
[edit protocols isis]
lab@r6# show traceoptions
file isis;
flag hello detail;
flag error detail;
```

The result of monitoring the isis trace file is shown in Listing 4.3:

Listing 4.3: IS-IS Tracing

```
[edit protocols isis]
lab@r6# run monitor start isis

[edit protocols isis]
lab@r6#
*** isis ***
May 17 17:24:43 Sending L1 LAN IIH on fe-0/1/0.0
May 17 17:24:43 max area 0, circuit type l1
May 17 17:24:43 No candidates for DR
May 17 17:24:43 hold time 27, priority 64, circuit id r6.02
May 17 17:24:43 speaks IP
May 17 17:24:43 speaks IPV6
May 17 17:24:43 IP address 10.0.8.5
May 17 17:24:43 area address 49.0002 (3)
May 17 17:24:44 Received L1 LAN IIH, source id 0100.0000.3005 on fe-0/1/0.0
May 17 17:24:44 intf index 2 addr 0.90.69.69.70.0, snpa 0:90:69:69:70:0
May 17 17:24:44 max area 0, circuit type l1, packet length 43
May 17 17:24:44 hold time 27, priority 64, circuit id 0100.0000.3005.02
May 17 17:24:44 speaks IP
May 17 17:24:44 speaks IPV6
```

```

May 17 17:24:44      IP address 10.0.8.6
May 17 17:24:44      area address 49.0001 (3)
May 17 17:24:44 ERROR: IIH from 0100.0000.3005 with no matching areas, interface
fe-0/1/0.0
May 17 17:24:44      local area 49.0002
May 17 17:24:44      remote area 49.0001 (3)
. . .

```

The trace output identifies the problem rather explicitly in this case. A level 1 router can only form an adjacency with routers in the same area, and r5 is identifying itself as belonging to area 49.0001. To resolve the problem, we reassign r5's NET using the following commands:

```

[edit interfaces lo0 unit 0 family iso]
lab@r5# delete address

[edit interfaces lo0 unit 0 family iso]
lab@r5# set address 49.0002.0100.0000.3005.00

```

The results are now confirmed:

```

[edit interfaces lo0 unit 0 family iso]
lab@r5# run show isis adjacency

```

Interface	System	L State	Hold (secs)	SNPA
as1.0	r4	2 Up	25	
at-0/2/1.35	r3	2 Up	22	
<u>fe-0/0/0.0</u>	<u>0100.0000.9006</u>	<u>1 New</u>	23	0:a0:c9:69:c5:27
<u>fe-0/0/1.0</u>	<u>r7</u>	<u>1 Up</u>	24	0:60:94:51:c4:27

Excellent! The level 1 adjacencies are now forming correctly. It should be noted that even though r5's NET identifies itself as belonging to the level 1 area 49.0002, its lo0 interface has not been configured as a level 1 interface. This is important, because doing so would cause the route to r5's loopback to be injected into the level 1 area in violation of the configuration requirements specified in this example.

Verify Multi-Level Operation

The proper operation of the level 1 area is confirmed with the following command, the results of which indicate the presence of the required default route to the L1/L2-attached router and the absence of level 2 router loopback addresses:

```

[edit protocols isis]
lab@r6# run show route protocol isis

inet.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

0.0.0.0/0          *[IS-IS/15] 00:03:58, metric 10, tag 1
                  > to 10.0.8.6 via fe-0/1/0.0
10.0.8.8/30       *[IS-IS/15] 00:03:58, metric 20, tag 1
                  > to 10.0.8.6 via fe-0/1/0.0
                  to 10.0.8.2 via fe-0/1/1.0
10.0.9.7/32       *[IS-IS/15] 00:55:25, metric 10, tag 1
                  > to 10.0.8.2 via fe-0/1/1.0

```

```
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

This author has found that issuing the following command on a L1/L2-attached router provides a quick check on the overall operation of an IS-IS network by confirming the presence of each IS-IS router and the setting of its SysID. The idea is similar to the verification of router LSAs on an OSPF ABR:

```
lab@r5> show isis hostname
```

```
IS-IS hostname database:
```

System Id	Hostname	Type
0100.0000.3003	r3	Dynamic
0100.0000.3004	r4	Dynamic
0100.0000.3005	r5	Static
0100.0000.9006	r6	Dynamic
0100.0000.9007	r7	Dynamic

As required, all five routers are present and the unique assignment of SysID values is confirmed. The last check is to verify connectivity to the loopback address of level 2 routers from within the level 1 area:

```
lab@r7> traceroute 10.0.3.3
```

```
traceroute to 10.0.3.3 (10.0.3.3), 30 hops max, 40 byte packets
```

```

 1  10.0.8.9 (10.0.8.9)  0.357 ms  0.258 ms  0.215 ms
 2  10.0.3.3 (10.0.3.3)  0.915 ms  1.124 ms  0.877 ms

```

```
lab@r7> traceroute 10.0.3.4
```

```
traceroute to 10.0.3.4 (10.0.3.4), 30 hops max, 40 byte packets
```

```

 1  10.0.8.9 (10.0.8.9)  0.330 ms  0.244 ms  0.218 ms
 2  10.0.3.4 (10.0.3.4)  0.445 ms  0.368 ms  0.355 ms

```

The traceroutes to both r3 and r4 succeed as we had hoped, confirming the correct operation of your multi-level IS-IS configuration.

IS-IS Adjacencies and IP Addressing

The IS-IS protocol carries IP routing information in an opaque manner using Type Length Values (TLVs) that act as extensions to the IS-IS routing protocol. This point is significant because the opacity of these IP-related TLVs will allow an IS-IS adjacency to form across a subnet with IP addressing problems. In general, most individuals are more familiar with the operation of OSPF than IS-IS, and such readers may find this behavior unexpected, as the OSPF protocol will not form an adjacency when IP parameters are mismatched. The key point is that IS-IS adjacency formation does not validate the interface IP-related configuration, and this can lead to some tricky troubleshooting problems as demonstrated here. In this example, r6 has had its IP address changed from the correct setting of 10.0.8.5/30 to the incorrect value of 10.0.10.5/30. This change does not affect IS-IS adjacency formation, and connectivity to r5's 10.0.8.6 address is still possible, albeit with a less-than-optimal route:

[edit]

```
lab@r6# show interfaces fe-0/1/0
unit 0 {
  family inet {
    address 10.0.10.5/30;
  }
  family iso;
}
```

[edit]

```
lab@r6# run show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
fe-0/1/0.0	<u>r5</u>	<u>1 Up</u>	22	0:90:69:69:70:0
fe-0/1/1.0	r7	1 Up	7	0:a0:c9:6f:7b:1a

[edit]

```
lab@r6# run traceroute 10.0.8.6
```

```
traceroute to 10.0.8.6 (10.0.8.6), 30 hops max, 40 byte packets
```

```
 1 10.0.8.2 (10.0.8.2) 0.197 ms 0.123 ms 0.096 ms
 2 10.0.8.6 (10.0.8.6) 0.488 ms 0.435 ms 0.410 ms
```

Though not required by the IS-IS specification or the IP-related RFCs, newer versions of JUNOS software have been modified to prevent the establishment of an IS-IS adjacency when IP parameters are mismatched. This modified behavior is nice, but should not be counted on because of the version-specific nature of this feature.

IS-IS Network Types

The IS-IS protocol offers support only for the broadcast and point-to-point network types. You will not be able to configure the IS-IS protocol on a WAN interface configured with the `multipoint` keyword, unless multicast support is also included:

```
[edit interfaces so-0/1/0]
lab@r4# show
dce;
hold-time up 30 down 30;
encapsulation frame-relay;
lmi {
    n392dce 2;
    n393dce 3;
    t392dce 25;
    lmi-type itu;
}
unit 100 {
    multipoint;
    family inet {
        address 10.0.2.6/30 {
            multipoint-destination 10.0.2.5 dlci 100;
        }
    }
    family iso;
}
```

The lack of a multicast DLCI assignment makes this an illegal configuration:

```
[edit interfaces so-0/1/0]
lab@r4# commit check
[edit interfaces so-0/1/0 unit 100 family iso]
    'family iso;'
    Family ISO not allowed on non-multicast-capable multipoint interfaces
DCD_PARSE_ERROR: config file parse failed
error: configuration check-out failed
```

At the time of this writing, carrier support for multicast Frame Relay and ATM circuits is virtually nonexistent. The grounding of the JNCIP examination in “real world” service provider-like environments means that you are unlikely to see multicast Frame Relay or ATM-related configuration requirements in the lab. With the `multicast-dlci` keyword present, the candidate configuration will commit but IS-IS will still treat the WAN link as a point-to-point interface.

The Designated Intermediate System

IS-IS makes use of the Designated Router (DR) concept for the same reasons as the OSPF protocol. Since IS-IS does not support NBMA network types, it elects only a DIS on the broadcast network type associated with LAN interfaces. Unlike OSPF, the DIS function in IS-IS is preemptive, which means that a router with a preferred priority setting will overthrow the existing DIS to claim its rightful position on the network. Should the DIS priority be the same, the tie is broken based on the MAC address (also called a Subnetwork Point of Attachment [SNPA]), with the router possessing the highest MAC address being declared the winner. Additional differences between IS-IS and OSPF include the fact that IS-IS does not support the concept of a BDR, and that LAN-attached stations will form a full mesh of IS-IS adjacencies, while in OSPF, such stations will form adjacencies only to the DR and BDR but not among DR/Other routers.

IS-IS Authentication

The IS-IS implementation offered in JUNOS software supports both simple password and HMAC-MD5–based authentication mechanisms. One key difference between OSPF and IS-IS authentication support relates to the ability to configure authentication for IS-IS hellos, LSP exchanges, or both.

Your next assignment is to add IS-IS authentication according to the criteria listed next. You should refer back to Figure 4.1 for topology details.

- Backbone area authenticates both hello and LSP exchanges using MD5 with a key value of *jni*.
- Area 49.0002 uses a plain text password of *jnx* for hello authentication only.
- No routing disruption can occur in the backbone when adding authentication.

Configure Authentication on *r5*

The following commands correctly configure the authentication parameters for area 49.0001. Note that the inclusion of the `no-authentication-check` option is required to ensure that adjacencies are not disrupted by the addition of authentication to the IS-IS level 2 backbone:

```
[edit protocols isis]
lab@r5# set no-authentication-check
```

```
[edit protocols isis]
lab@r5# set level 2 authentication-type md5 authentication-key jni
```

The next set of commands adds the required hello authentication to each of *r5*'s level 2 interfaces. Authentication is not configured on interface `lo0` because LSP flooding and adjacency formation do not occur on this virtual interface:

```
[edit protocols isis]
lab@r5# set interface as1.0 hello-authentication-type md5
```

```
[edit protocols isis]
lab@r5# set interface as1.0 hello-authentication-key jni

[edit protocols isis]
lab@r5# set interface at-0/2/1.35 hello-authentication-type md5

[edit protocols isis]
lab@r5# set interface at-0/2/1.35 hello-authentication-key jni
```

Next, simple password authentication, as required for area 49.0002 hello exchanges, is configured on r5:

```
[edit protocols isis]
lab@r5# set interface fe-0/0/0 hello-authentication-type simple

[edit protocols isis]
lab@r5# set interface fe-0/0/0 hello-authentication-key jnx

[edit protocols isis]
lab@r5# set interface fe-0/0/1 hello-authentication-type simple

[edit protocols isis]
lab@r5# set interface fe-0/0/1 hello-authentication-key jnx
```

The modified configuration for r5 is shown next with the newly added authentication configuration highlighted:

```
[edit protocols isis]
lab@r5# show
no-authentication-check;
level 2 {
  authentication-key "$9$iqPQB1hSrv"; # SECRET-DATA
  authentication-type md5; # SECRET-DATA
}
interface fe-0/0/0.0 {
  hello-authentication-key "$9$VhsqJTQn6A0"; # SECRET-DATA
  hello-authentication-type simple; # SECRET-DATA
  level 2 disable;
}
interface fe-0/0/1.0 {
  hello-authentication-key "$9$iqPQB1hc1M"; # SECRET-DATA
  hello-authentication-type simple; # SECRET-DATA
```

```

    level 2 disable;
}
interface at-0/2/1.35 {
    hello-authentication-key "$9$p-Xm0IcN-wYgJ"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface as1.0 {
    hello-authentication-key "$9$gXaGi6/tuOR"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}

```

After committing the authentication changes on r5, we confirm that all level 2 adjacencies remain in place, despite the absence of compatible authentication settings on the remaining routers that make up the test bed:

```

[edit]
lab@r5# commit and-quit
commit complete
Exiting configuration mode

lab@r5> show isis adjacency
Interface           System      L State      Hold (secs) SNPA
as1.0                r4          2 Up         19
at-0/2/1.35         r3          2 Up         21
fe-0/0/0.0          r6          1 Up          8 0:a0:c9:69:c5:27
fe-0/0/1.0          r7          1 Up          23 0:60:94:51:c4:27

```

Configure Authentication on r3 and r4

The authentication configuration for r3 and r4 is similar to that of r5. The only difference is the lack of area 49.0002 and level 1–related authentication, which is not required because these routers are only level 2–attached. The completed configuration for the remaining level 2 routers is shown here, starting with r3:

```

[edit protocols isis]
lab@r3# show
no-authentication-check;

```

```

level 2 {
    authentication-key "$$nhTp9tOMWxNds"; # SECRET-DATA
    authentication-type md5; # SECRET-DATA
}
interface fe-0/0/2.0 {
    passive;
    level 1 disable;
}
interface at-0/1/0.35 {
    hello-authentication-key "$$wx2oGzF/Ct0"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface so-0/2/0.100 {
    hello-authentication-key "$$yuZeMXaJDikP"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}

```

r4's configuration is very similar to that of r3:

```

[edit protocols isis]
lab@r4# show
no-authentication-check;
level 2 {
    authentication-key "$$XX4NVYq.5QF/"; # SECRET-DATA
    authentication-type md5; # SECRET-DATA
}
interface so-0/1/0.100 {
    hello-authentication-key "$$0QpxIhrVb24aU"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface as0.0 {
    hello-authentication-key "$$SCLlvLoaUjHm"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}

```

```
interface lo0.0 {
    level 1 disable;
}
```

Configure Authentication in Area 49.0002

The following commands correctly configure r7 for simple password-based authentication of hello messages. The use of `no-authentication-check` is not required in this area, so we expect to see the loss of adjacency to r6 when the changes are committed. The adjacency to r5, however, should remain operational because it's already being configured with compatible authentication parameters:

```
[edit protocols isis]
lab@r7# set interface all hello-authentication-type simple
```

```
[edit protocols isis]
lab@r7# set interface all hello-authentication-key jnx
```

```
[edit protocols isis]
lab@r7# commit
commit complete
```

```
[edit protocols isis]
lab@r7# run show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
fe-0/3/0.0	r6	1 Rejected	16	0:a0:c9:69:c1:e4
fe-0/3/1.0	r5	1 Up	7	0:90:69:69:70:1

Just as predicted, the lack of compatible hello authentication in r6 results in the rejection of its hello packets. This situation will clear up when r6 is compatibly configured, as shown next:

```
[edit protocols isis]
lab@r6# show
level 2 disable;
interface all {
    hello-authentication-key "$9$RTtcrvY2aJdk"; # SECRET-DATA
    hello-authentication-type simple; # SECRET-DATA
}
```

The authentication changes are committed on r6, and the results are confirmed:

```
[edit protocols isis]
lab@r6# commit
commit complete
```

```
[edit protocols isis]
lab@r6# run show isis adjacency
Interface          System      L State      Hold (secs) SNPA
fe-0/1/0.0         r5          1 Up         21 0:90:69:69:70:0
fe-0/1/1.0         r7          1 Initializing 22 0:a0:c9:6f:7b:1a
```

```
[edit protocols isis]
lab@r6# run show isis adjacency
Interface          System      L State      Hold (secs) SNPA
fe-0/1/0.0         r5          1 Up         22 0:90:69:69:70:0
fe-0/1/1.0         r7          1 Up         26 0:a0:c9:6f:7b:1a
```

Troubleshoot IS-IS Authentication

IS-IS authentication problems often result in the same types of neighbor detection and adjacency failures that can also be attributed to numerous problems at the IS-IS, link, or physical layers. The operator can quickly isolate physical and link layer problems through ping testing to a directly connected router. Authentication-related problems should be suspected when ping testing succeeds in the face of IS-IS adjacency failures. Adding the `no-authentication-check` option can quickly confirm if authentication is causing an adjacency problem because this causes the receiving router to ignore the authentication parameters (or lack thereof) in the IS-IS PDUs it receives.

Often, the use of `monitor traffic` will make authentication problems obvious. The following example shows a capture taken from `r7`'s `fe-0/3/0` interface before `r6` has been correctly configured for hello authentication:

```
[edit protocols isis]
lab@r7# run monitor traffic interface fe-0/3/0
Listening on fe-0/3/0
19:55:18.586402 Out iso isis 0:a0:c9:6f:7b:1a > 1:80:c2:0:0:14 len=49
L1 lan iih, circuit 11 only, holding time 27
source 1:0:0:0:90:7, length 49
lan id 1:0:0:0:90:7(2)
Supports protocols are CC 8E
IP address: 10.0.8.2
area addresses
49.0002 (3)
authentication data
016a 6e78
```

```

19:55:20.195900 In iso isis 0:a0:c9:69:c1:e4 > 1:80:c2:0:0:14 len=1492
                L1 lan iih, circuit 11 only, holding time 27
                source 1:0:0:0:90:6, length 1492
                lan id 1:0:0:0:90:6(3)
                neighbor addresses
                0:a0:c9:6f:7b:1a
                Supports protocols are CC 8E
                IP address: 10.0.8.1
                area addresses
                49.0002 (3)
                padding for 255 bytes
                packet exceeded snapshot

```

This decode clearly shows that simple password authentication is present in the hello packets sent out r7's fe-0/3/0 interface. The plain text password value of *jnx* is displayed in hexadecimal form with 6a 6e 78 being ASCII code for the configured *jnx* key. In contrast, the hello packets received by r7 clearly lack the authentication field.

The following capture, taken from r5's aggregated SONET interface, illustrates the difference between the simple password and MD5-based hello packet authentication approaches:

```

[edit protocols isis]
lab@r5# run monitor traffic interface as1
Listening on as1
. . .
05:03:53.746131 In iso isis len=72
                PTP iih, circuit 12 only, holding time 27
                source 1:0:0:0:30:4, length 72
                PTP adjacency status UP
                Supports protocols are CC 8E
                IP address: 10.0.2.10
                area addresses
                49.0001 (3)
                authentication data
                3661 755c 5532 ce85 645f a7d0 8b92 8934 69

```

Once again, it is clear that the received hello packet is carrying some type of authentication information. However, it is not obvious whether the packets are using simple password or MD5-based secrets. If you suspect that mismatched MD5 secrets have been configured, you will need to reset the MD5 key in all routers. Reverse engineering the MD5 hash is not possible, even with an ASCII code chart.

Protocol Tracing

While monitoring traffic has the benefit of being fast and dirty, the output will often leave much to the operator's interpretation, which is in stark contrast to the overt error messages that are sometimes seen with protocol tracing. The following tracing configuration has been added to r7:

```
[edit protocols isis]
lab@r7# show traceoptions
file isis;
flag error detail;
flag hello detail;
```

This tracing configuration produced the following output, which makes it hard to miss the authentication-related nature of the problem:

```
[edit]
lab@r7# run monitor start isis

lab@r7#
*** isis ***
May 17 20:08:57 Received L1 LAN IIH, source id r6 on fe-0/3/0.0
May 17 20:08:57     intf index 2 addr 0.a0.c9.69.c1.e4, snpa 0:a0:c9:69:c1:e4
May 17 20:08:57     max area 0, circuit type 11, packet length 1492
May 17 20:08:57     hold time 27, priority 64, circuit id r6.03
May 17 20:08:57     neighbor 0:a0:c9:6f:7b:1a (ourselves)
May 17 20:08:57     speaks IP
May 17 20:08:57     speaks IPV6
May 17 20:08:57     IP address 10.0.8.1
May 17 20:08:57     area address 49.0002 (3)
May 17 20:08:57 ERROR: IIH from r6 without authentication
May 17 20:08:57 ERROR: previous error from L1, source r6 on fe-0/3/0.0
```



Real World Scenario

How Can You Confirm Area 49.0001 Authentication Settings?

The presence of the `no-authentication-check` option in the backbone routers can cause incompatible or incorrect authentication parameters to go undetected. Before moving on, you should temporarily delete this option on a backbone router and verify that all level 2 adjacencies are maintained. After confirming that all is well, this option should be reinstated, as its presence is needed to show compliance with the specified configuration requirements.

```
[edit protocols isis]
lab@r5# delete no-authentication-check

[edit protocols isis]
lab@r5# commit
commit complete

[edit protocols isis]
lab@r5# run show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
as1.0	r4	2 Up	23	
at-0/2/1.35	r3	2 Up	23	
fe-0/0/0.0	r6	1 Up	8	0:a0:c9:69:c5:27
fe-0/0/1.0	r7	1 Up	20	0:60:94:51:c4:27

```
[edit protocols isis]
lab@r5# set no-authentication-check
[edit protocols isis]
lab@r5# commit
commit complete
```

IS-IS Policy

JUNOS software policy is used with IS-IS to configure route leaking, route summarization, and route redistribution. The configuration examples in this section demonstrate the use and verification of IS-IS routing policy.

Route Leaking and Summarization

Route leaking allows the injection of routing information from the IS-IS backbone into a level 1 area. The added routing information in the level 1 area reduces the area's reliance on the default

route generated by level 2–attached routers, and can eliminate the potential for inefficient routing that can result from using the metrically closest attached router for all inter-area destinations.

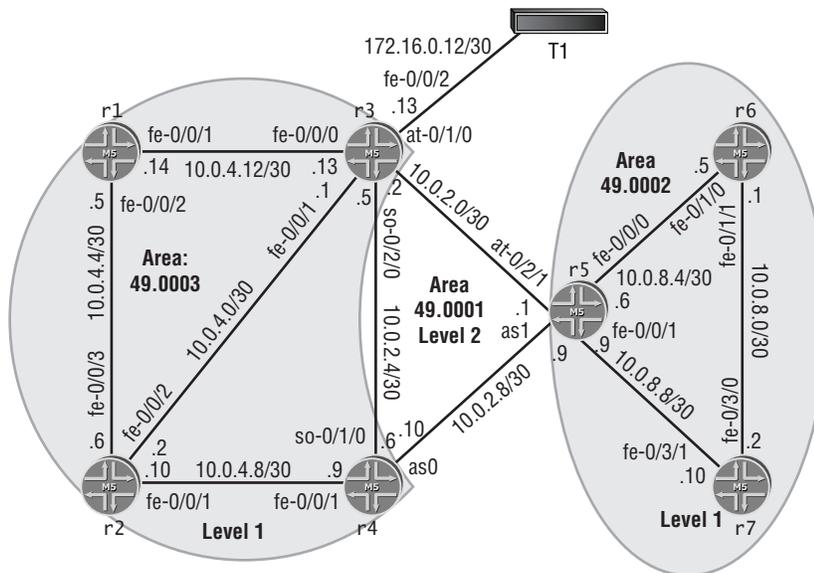
To complete this configuration example, you will need to configure area 49.0003 and then apply routing policy to effect route leaking and route summarization for the area according to the following criteria:

- Leak level 2 loopback addresses into area 49.0003 and ensure optimal routing to these prefixes.
- Summarize area 49.0003 addresses into the backbone.
- Maintain connectivity between all router loopback addresses.

Area 49.0003 Initial Configuration

Before route leaking and summarization can be demonstrated, you must first correctly configure area 49.0003 as a level 1 area according to Figure 4.2.

FIGURE 4.2 Route leaking and summarization



To complete the initial configuration of area 49.003, you will need to reassign the NET on both r3 and r4 (the current NET will prevent level 1 adjacency formation in area 49.0003), and configure the correct interfaces for level 1 operation. A working configuration from r4 is shown here with highlighted changes:

```
[edit]
lab@r4# show interfaces fe-0/0/1
speed 100m;
link-mode half-duplex;
```

```

fastether-options {
    no-flow-control;
}
unit 0 {
    family inet {
        address 10.0.4.9/30;
    }
    family iso;
}

```

```

[edit]
lab@r4# show interfaces lo0
unit 0 {
    family inet {
        address 10.0.3.4/32;
    }
    family iso {
        address 49.0003.0100.0000.3004.00;
    }
}

```

The modified IS-IS stanza on r4 includes the declaration of interface fe-0/0/1 as a level 1-only interface:

```

[edit]
lab@r4# show protocols isis
no-authentication-check;
level 2 {
    authentication-key "$$XX4NVYq.5QF/"; # SECRET-DATA
    authentication-type md5; # SECRET-DATA
}
interface fe-0/0/1.0 {
    level 2 disable;
}
interface so-0/1/0.100 {
    hello-authentication-key "$$0QpxIhrVb24aU"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface as0.0 {
    hello-authentication-key "$$SCL1vLoaUjHm"; # SECRET-DATA

```

```

    hello-authentication-type md5; # SECRET-DATA
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}

```

A functional configuration for a level 1 router in area 49.0003 is shown next for r1 with IS-IS additions highlighted:

```

[edit]
lab@r1# show interfaces fe-0/0/1
vlan-tagging;
unit 200 {
    vlan-id 200;
    family inet {
        address 10.0.4.14/30;
    }
    family iso;
}

```

```

[edit]
lab@r1# show interfaces fe-0/0/2
fastether-options {
    source-filtering;
    source-address-filter {
        00:a0:c9:6f:70:0d;
    }
}
unit 0 {
    family inet {
        address 10.0.4.5/30;
    }
    family iso;
}

```

```

[edit]
lab@r1# show interfaces lo0
unit 0 {
    family inet {
        address 10.0.6.1/32;
    }
}

```

```

    family iso {
      address 49.0003.0100.0000.6001.00;
    }
}

```

r1's IS-IS stanza correctly configures it as a level 1-only router:

```

[edit]
lab@r1# show protocols isis
level 2 disable;
interface all;

```

Area 49.0003 Verification

After completing the initial configuration for area 49.0003, you can quickly assess your work by verifying that the appropriate adjacencies and SysIDs are present in area 49.0003:

```

lab@r3> show isis adjacency

```

Interface	System	L State	Hold (secs)	SNPA
at-0/1/0.35	r5	2 Up	21	
<u>fe-0/0/0.0</u>	<u>r1</u>	<u>1 Up</u>	7	0:a0:c9:6f:7b:3e
<u>fe-0/0/1.0</u>	<u>r2</u>	<u>1 Up</u>	26	0:a0:c9:6f:7a:ff
so-0/2/0.100	r4	2 Up	26	

```

lab@r3> show isis hostname
IS-IS hostname database:

```

System Id	Hostname	Type
0100.0000.3003	r3	Static
0100.0000.3004	r4	Dynamic
0100.0000.3005	r5	Dynamic
<u>0100.0000.6001</u>	<u>r1</u>	Dynamic
<u>0100.0000.6002</u>	<u>r2</u>	Dynamic

Configure Route Leaking

For this example, your IS-IS route-leaking configuration is performed on the L1/L2-attached routers that serve area 49.0003. You must use a policy that makes use of route filters to achieve the goal of leaking only the backbone area's loopback addresses. Before route leaking is configured, you first analyze the state of the routing table for level 1 routers in area 49.0003:

```

lab@r2> show route protocol isis

```

inet.0: 15 destinations, 15 routes (15 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

0.0.0.0/0      *[IS-IS/15] 01:54:13, metric 10, tag 1
                to 10.0.4.9 via fe-0/0/1.0
                > to 10.0.4.1 via fe-0/0/2.0
10.0.4.12/30  *[IS-IS/15] 01:54:21, metric 20, tag 1
                to 10.0.4.1 via fe-0/0/2.0
                > to 10.0.4.5 via fe-0/0/3.0
10.0.6.1/32   *[IS-IS/15] 01:54:28, metric 10, tag 1
                > to 10.0.4.5 via fe-0/0/3.0

```

From this display, we can see that backbone area loopback addresses are not yet present in area 49.0003. The following commands create the policy needed to correctly leak lo0 prefixes into area 49.0003:

```

[edit policy-options policy-statement leak]
lab@r4# set term 1 from protocol isis

```

```

[edit policy-options policy-statement leak]
lab@r4# set term 1 from level 2

```

```

[edit policy-options policy-statement leak]
lab@r4# set term 1 from route-filter 10.0.3/24 longer

```

```

[edit policy-options policy-statement leak]
lab@r4# set term 1 to level 1

```

```

[edit policy-options policy-statement leak]
lab@r4# set term 1 then accept

```

The completed policy is now displayed:

```

[edit policy-options policy-statement leak]
lab@r4# show
term 1 {
  from {
    protocol isis;
    level 2;
    route-filter 10.0.3.0/24 longer;
  }
  to level 1;
  then accept;
}

```

The route-leaking policy is now applied as an export policy to the IS-IS instance and the changes are committed:

```
[edit]
lab@r4# set protocols isis export leak
```

```
[edit]
lab@r4# commit
commit complete
```

Verify Route Leaking

The effects of your route-leaking policy are tested with the following commands:

```
lab@r2> show route protocol isis

inet.0: 17 destinations, 17 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[IS-IS/15] 01:57:53, metric 10, tag 1
                   to 10.0.4.9 via fe-0/0/1.0
                   > to 10.0.4.1 via fe-0/0/2.0
10.0.3.3/32       *[IS-IS/18] 00:00:54, metric 20, tag 1
                   > to 10.0.4.9 via fe-0/0/1.0
10.0.3.5/32       *[IS-IS/18] 00:00:54, metric 20, tag 1
                   > to 10.0.4.9 via fe-0/0/1.0
10.0.4.12/30      *[IS-IS/15] 01:58:01, metric 20, tag 1
                   to 10.0.4.1 via fe-0/0/2.0
                   > to 10.0.4.5 via fe-0/0/3.0
10.0.6.1/32       *[IS-IS/15] 01:58:08, metric 10, tag 1
                   > to 10.0.4.5 via fe-0/0/3.0
```

While these results are promising, r4's loopback address is absent in area 49.0003. This situation is caused by the policy being written to match on IS-IS level 2 routes coupled with the fact that r4 has learned about its own lo0 address from the direct protocol. This problem is easily rectified with the highlighted policy modification on r4:

```
[edit policy-options policy-statement leak]
lab@r4# show
term 1 {
  from {
    protocol isis;
    level 2;
    route-filter 10.0.3.0/24 longer;
  }
}
```

```

    }
    to level 1;
    then accept;
}
term 2 {
  from {
    protocol direct;
    route-filter 10.0.3.0/24 longer;
  }
  to level 1;
  then accept;
}
}

```

Although the testing conducted with the 5.2 version of JUNOS software used in this test bed indicates that the `to level 1` criteria is not needed for the route leaking behavior specified in this example, such syntax is needed when the goal is to leak a prefix into either level 1 *or* level 2, but not into both levels 1 and 2. The `to level 1` syntax has been included here because it helps to reinforce the readers' awareness of route-leaking directionality. Also of note is the fact that this policy does not make use of a "reject all" term, which means the default IS-IS policy of leaking level 1 prefixes into level 2 has not been altered. The net result is the expectation that r4 will continue to leak level 1 routes into level 2 while also leaking specific level 2 prefixes into area 49.0003.

To complete the route-leaking configuration, a similar policy must be applied to r3. This will ensure that the suboptimal routing from area 49.0003 to backbone router loopback addresses, which is demonstrated next, does not occur:

```

lab@r1> traceroute 10.0.3.3
traceroute to 10.0.3.3 (10.0.3.3), 30 hops max, 40 byte packets
 1 10.0.4.6 (10.0.4.6) 0.219 ms 0.167 ms 0.115 ms
 2 10.0.4.9 (10.0.4.9) 0.353 ms 0.307 ms 0.286 ms
 3 10.0.3.3 (10.0.3.3) 0.504 ms 0.433 ms 0.421 ms

```

The extra hops in this capture are the result of having only one of the attached routers perform the route leaking. After applying the same route-leaking policy to r3, we see that optimal routing to level 2 lo0 addresses has been achieved:

```

lab@r1> traceroute 10.0.3.3
traceroute to 10.0.3.3 (10.0.3.3), 30 hops max, 40 byte packets
 1 10.0.3.3 (10.0.3.3) 0.513 ms 0.376 ms 0.346 ms

```

Configure Route Summarization

Because the IS-IS protocol does not offer support of area-range statements, route summarization requires the definition of an aggregate route and the policy needed to advertise this aggregate while suppressing the more specific prefixes. Before configuring summarization, analyze the route

table of a backbone router with regard to area 49.0003 routes in order to better judge the effect of your route summarization configuration:

```
lab@r5> show route protocol isis 10.0.4/22
```

```
inet.0: 24 destinations, 24 routes (24 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
10.0.4.0/30      *[IS-IS/18] 02:46:47, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.4.4/30      *[IS-IS/18] 00:02:27, metric 30, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
                 to 10.0.2.10 via as1.0
10.0.4.8/30      *[IS-IS/18] 02:48:01, metric 20, tag 2
                 > to 10.0.2.10 via as1.0
10.0.4.12/30     *[IS-IS/18] 02:46:47, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.6.1/32      *[IS-IS/18] 00:02:34, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.6.2/32      *[IS-IS/18] 00:02:27, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
                 to 10.0.2.10 via as1.0
```

These results confirm that summarization into the backbone does not occur by default. You can configure r3 to advertise a summary, and to suppress the more specific level 1 routes, with the following policy and aggregate route definition:

```
[edit]
```

```
lab@r3# show policy-options policy-statement summ
```

```
term 1 {
    from protocol aggregate;
    to level 2;
    then accept;
}
term 2 {
    from {
        protocol isis;
        level 1;
    }
    to level 2;
    then reject;
}
```

Term 2 in this policy negates the default IS-IS policy of leaking level 1 routes into level 2, which is intended to prevent the advertisement of the specific prefixes in area 49.0003. The aggregate route for area 49.0003 prefixes is displayed next:

```
[edit]
lab@r3# show routing-options aggregate
route 10.0.4.0/22;
```

The *summ* policy is then applied to r3's IS-IS instance as an export policy:

```
[edit]
lab@r3# show protocols isis export
export [ leak summ ];
```

Verify Route Summarization

You can easily verify the results of your route summarization by once again displaying the contents of a backbone router's routing table, as it relates to area 49.0003 routes:

```
lab@r5> show route protocol isis 10.0.4/22
```

```
inet.0: 25 destinations, 25 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.4.0/22      *[IS-IS/165] 00:00:22, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.4.0/30      *[IS-IS/18] 03:04:49, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.4.4/30      *[IS-IS/18] 00:00:22, metric 30, tag 2
                 > to 10.0.2.10 via as1.0
10.0.4.8/30      *[IS-IS/18] 00:02:44, metric 20, tag 2
                 > to 10.0.2.10 via as1.0
10.0.4.12/30     *[IS-IS/18] 03:04:49, metric 20, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
10.0.6.1/32      *[IS-IS/18] 00:00:22, metric 30, tag 2
                 > to 10.0.2.10 via as1.0
10.0.6.2/32      *[IS-IS/18] 00:00:22, metric 20, tag 2
                 > to 10.0.2.10 via as1.0
```

You can also assess the operation of the summarization policy by examining the contents of r3's LSPs:

```
lab@r5> show isis database r3 detail
IS-IS level 1 link-state database:
```

IS-IS level 2 link-state database:

```
r3.00-00 Sequence: 0x93, Checksum: 0x442, Lifetime: 413 secs
IS neighbor:                r4.00 Metric:    10
IS neighbor:                r5.00 Metric:    10
IP prefix:                   10.0.4.0/22 Metric:    10 External
IP prefix:                   10.0.2.4/30 Metric:    10 Internal
IP prefix:                   10.0.2.0/30 Metric:    10 Internal
IP prefix:                   10.0.4.0/30 Metric:    10 Internal
IP prefix:                   10.0.4.12/30 Metric:    10 Internal
IP prefix:                   10.0.3.3/32 Metric:     0 Internal
IP prefix:                   172.16.0.12/30 Metric:    10 Internal
```

These results confirm that r3 is correctly advertising the aggregate in its level 2 LSP, but also indicate that the 10.0.4.0/30 and 10.0.12.0/30 contributing routes are being incorrectly leaked (as indicated by the entries using at-0/2/1.35 as a next hop in r5's routing table and by their presence in r3's level 2 LSP). As you might expect, the problem here relates to the summarization policy, and the fact that it is not rejecting routes for directly connected interfaces that have IS-IS configured. Displaying one of these routes on r3 confirms this is indeed the case:

[edit]

lab@r3# **run show route 10.0.4.12/30**

```
inet.0: 26 destinations, 26 routes (26 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.4.12/30      *[Direct/0] 2d 03:50:53
                  > via fe-0/0/0.0
10.0.4.13/32     *[Local/0] 2d 04:16:09
                  Local via fe-0/0/0.0
```

To correct this problem, the summarization policy on r3 is modified as highlighted next. This is just one example of many possible policy modifications that could have resolved the route summarization problem:

[edit policy-options policy-statement summ]

lab@r3# **show**

```
term 1 {
    from protocol aggregate;
    to level 2;
    then accept;
}
term 2 {
```

```

    from {
      route-filter 10.0.4.0/22 longer;
    }
    to level 2;
    then reject;
  }

```

In this example, omitting the `to level 2` condition in term 2 would cause problems in area 49.0003 because routes matching the route-filter would then not be injected into any IS-IS level, and these routes should be present in the level 1 area. After applying the same summarization policy to r4, the results are once again verified on the backbone router r5:

```
lab@r5> show route 10.0.4/22
```

```

inet.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.4.0/22          *[IS-IS/165] 00:00:00, metric 20, tag 2
                    to 10.0.2.2 via at-0/2/1.35
                    > to 10.0.2.10 via as1.0

```

Very good, the summary route for area 49.0003 now has two viable next hops, and the more specific prefixes are no longer being injected into the level 2 area in accordance with the configuration requirements.

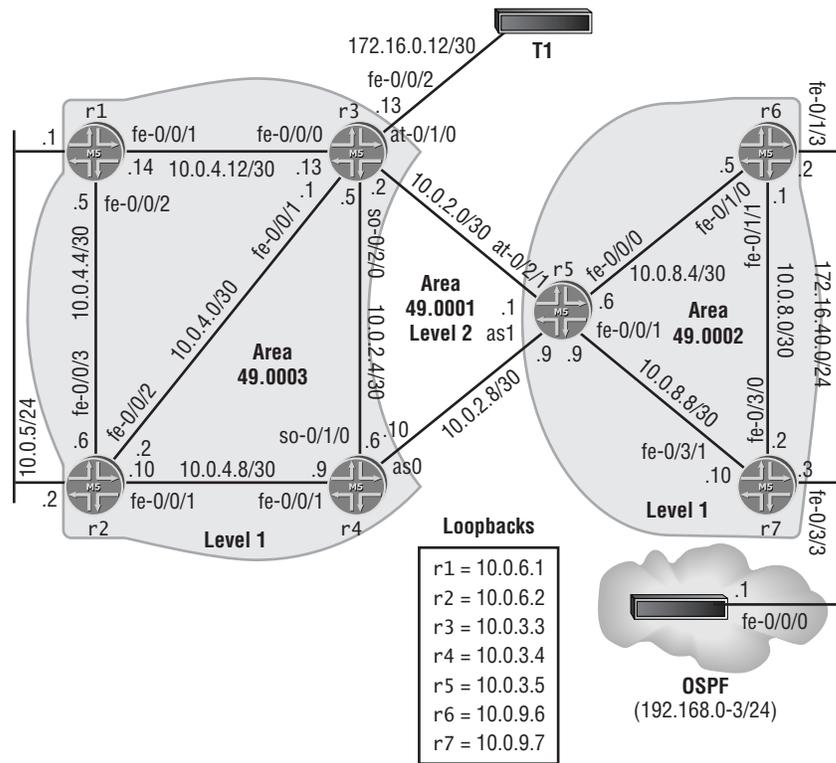
Route Redistribution

Export policy can also be applied to IS-IS to facilitate route redistribution and the manipulation of metrics at L1/L2-attached routers. JUNOS software does not support the application of import policy for link-state routing protocols like IS-IS because such policies could lead to inconsistent LSDB entries, which in turn could result in routing inconsistencies. The filtering of LSPs is only possible in a multi-level topology, which will suppress the injection of level 2 LSPs into level 1 areas by default.

The use of routing policy to evoke route redistribution has already been demonstrated several times in this book, as has the use of policy to set the metric values for a redistributed static route. The following configuration scenario will demonstrate these capabilities as part of an IS-IS and OSPF mutual route redistribution scenario.

You will now add to the complexity of your network by configuring mutual route redistribution between IS-IS and OSPF as shown in Figure 4.3. The configuration criteria for this example follow the figure.

FIGURE 4.3 IS-IS route redistribution



- Redistribute the 10.0.5/25 subnet from both r1 and r2 so that level 2 routers display a route metric of at least 500.
- r5 must be able to load balance to the 10.0.5/24 prefix.
- Redistribute the IS-IS default route from both r6 and r7 to the OSPF router.
- Redistribute OSPF routes 192.168.0/24 through 192.168.3/24 into area 49.0002 from both r6 and r7.
- Ensure that the OSPF router, r6, and r7 can reach destinations on the 10.0.5/24 subnet.
- Except for r5, no single point of failure can isolate the OSPF router from reaching the 10.0.5/24 subnet.
- Continue to advertise the area 49.0003 summary while suppressing more specific prefixes, except for the 10.0.5/24 route in accordance with the first requirement.

The OSPF Router's Configuration

The pertinent portions of the OSPF router's configuration are shown and described in the following. You may not modify the OSPF router's configuration in this configuration example:

```
[edit]
lab@ospf# show protocols
ospf {
    export ospf;
    area 0.0.0.1 {
        interface fe-0/0/0.0;
    }
}
```

The OSPF stanza defines OSPF area 1 operation with an export policy called *ospf*. The contents of this policy are as follows:

```
[edit]
lab@ospf# show policy-options
policy-statement ospf {
    term 1 {
        from protocol static;
        then accept;
    }
}
```

The *ospf* policy will redistribute routes learned through the static protocol into OSPF as AS externals. The OSPF router's *routing-options* stanza contains the various 192.168.x static route definitions, with their next hops set to discard in this example:

```
[edit]
lab@ospf# show routing-options
static {
    route 192.168.0.0/24 discard;
    route 192.168.1.0/24 discard;
    route 192.168.2.0/24 discard;
    route 192.168.3.0/24 discard;
}
```

The OSPF router's interface configuration correctly sets the addressing for the OSPF subnet, and assigns host addresses from each of the 192.168.x static routes to the lo0 interface so that pings and traceroutes to host-id 1 on each 192.168.x subnet can complete successfully:

```
[edit]
lab@ospf# show interfaces
```

```

. . .
fe-0/0/0 {
    unit 0 {
        family inet {
            address 172.16.40.1/24;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.1/32;
            address 192.168.1.1/32;
            address 192.168.2.1/32;
            address 192.168.3.1/32;
        }
    }
}
}

```

IS-IS Redistribution on *r1* and *r2*

Your configuration begins by creating and applying a policy that will redistribute the 10.0.5/24 direct route from *r1* and *r2* into IS-IS with a metric of at least 500. The following policy will work for the requirements of this example, but other techniques are certainly possible:

```

[edit policy-options policy-statement direct]
lab@r1# show
term 1 {
    from {
        route-filter 10.0.5.0/24 exact;
    }
    then {
        metric 500;
        accept;
    }
}

```

The policy is applied to *r1*'s IS-IS instance as an export policy, and the effects are confirmed on *r5*:

```

[edit]
lab@r5# run show route 10.0.5/24

```

```
[edit]
lab@r5#
```

The route's absence on r5 is not a good sign. Perhaps we should look closer to the route's source by examining r3's view of the situation?

```
[edit]
lab@r3# run show route 10.0.5/24
```

```
inet.0: 27 destinations, 27 routes (27 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.5.0/24      *[IS-IS/160] 00:02:51, metric 73, tag 1
                  > to 10.0.4.14 via fe-0/0/0.0
```

Troubleshoot IS-IS Redistribution

The 10.0.5/24 route is present on r3, so r1's export policy must be doing something. The metric value of 73 is odd (both literally and in the context of this example), considering that the policy used to redistribute the route was configured to set the metric to 500. When faced with multiple symptoms such as these, this author would suggest that you prioritize the symptoms so that you can concentrate on resolving the most significant problems first. In this case, it is suggested that you resolve the route's absence on r5 before worrying about the incorrect metric, because in many cases connectivity problems will result in more point losses on your exam, which makes the missing route your highest priority. Besides, it is often the case that multiple symptoms will result from a common problem, so tackling the most significant issue first may result in less work overall.

The route's presence in r3 combined with its absence in r3's L2 LSP points to routing policy as the probable cause. There are really two issues at work here; one is general in nature while the other is somewhat specific to this example. The general problem relates to the fact that the default IS-IS export policy will not send level 1 external routes into the IS-IS backbone. The specific issue lies in the route-filter statement used in the existing *sum* policy that was designed to suppress the specific routes associated with area 49.0003, as the 10.0.5/24 route happens to fall within the aggregate 10.0.4/22, and is therefore being explicitly rejected. The problem area in the *sum* policy is highlighted next:

```
. . .
term 2 {
  from {
    route-filter 10.0.4.0/22 longer;
  }
  to level 2;
  then reject;
}
```

The highlighted policy modification punches a hole in term 2 to explicitly allow the 10.0.5/24 prefix:

```
[edit policy-options policy-statement summ]
lab@r3# show
term 1 {
    from protocol aggregate;
    to level 2;
    then accept;
}
term 2 {
    from {
        route-filter 10.0.4.0/22 longer;
        route-filter 10.0.5.0/24 exact accept;
    }
    to level 2;
    then reject;
}
```

The use of an `accept` terminating action in conjunction with the 10.0.5/24 route-filter statement causes packets that match the route-filter exactly to be accepted while all other area 49.0003 prefixes are rejected by the terminating action associated with term 2. After committing the modification to the `summ` policy on r3, the results are once again confirmed on r5:

```
[edit]
lab@r5# run show route 10.0.5/24

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.5.0/24      *[IS-IS/165] 00:00:05, metric 73, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
```

Good! The route is now present on r5, but the metric is still stuck at 73. This is a tricky problem to solve, as its solution depends on an in-depth understanding of the IS-IS protocol and JUNOS software behavior. The issue here relates to narrow vs. wide metrics, and the fact that the former supports a maximum link cost of 63 and a total path cost of 1023. The use of narrow metrics explains why the route's metric is stuck at 73. Neither r1 nor r3 can advertise a link metric higher than 63, and the cost for r5 to reach r3 is 10, which results in the displayed metric value of 73. By default, JUNOS software sends both wide and narrow metrics along with each route, and when given the choice, Juniper Networks routers will opt to use the narrow metric. To

resolve this situation, you must modify the routers in areas 49.0003 and 49.0001 to use wide metrics only, with the following command:

```
[edit]
lab@r1# set protocols isis level 1 wide-metrics-only
```

After the change has been committed, the contents of r1's level 1 LSP is verified:

```
[edit]
lab@r1# run show isis database r1 detail
IS-IS level 1 link-state database:

r1.00-00 Sequence: 0x70, Checksum: 0x75bc, Lifetime: 1195 secs
IS neighbor:                r1.03 Metric:    10
IS neighbor:                r1.02 Metric:    10
IP prefix:                   10.0.4.4/30 Metric:    10 Internal
IP prefix:                   10.0.4.12/30 Metric:    10 Internal
IP prefix:                   10.0.6.1/32 Metric:    0 Internal
IP prefix: 10.0.5.0/24 Metric:    500 Internal
```

. . .

The following command is now applied to r3 to enable its use of wide metrics only, in the level 2 area:

```
[edit]
lab@r3# set protocols isis level 2 wide-metrics-only
```

After the change is committed, the route is once again analyzed on r5:

```
[edit]
lab@r5# run show route 10.0.5/24

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.5.0/24          *[IS-IS/18] 00:00:00, metric 520, tag 2
                    > to 10.0.2.2 via at-0/2/1.35
```

Good! Backbone routers now see a metric greater than 500, in accordance with the specified requirements. To complete this example, you must now apply the same IS-IS export policy to r2, and you must also set wide metrics only, on r2 and r4, as shown for r1 and r3. The added configuration is needed to comply with the redundancy requirements of this example. After making

the changes, we expect to see two viable next hops for the 10.0.5/24 prefix on r5. Failing to disable narrow metrics on r4 will result in a single next hop on r5 (and point loss on your exam for not allowing load-balancing), because r5 will prefer the narrow metric sent from r4 to the wide metric that is advertised by r3. This situation is demonstrated in the following capture:

```
[edit]
lab@r5# run show route 10.0.5/24

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.5.0/24      *[IS-IS/18] 00:00:02, metric 73, tag 2
                 > to 10.0.2.10 via as1.0
```

You now correctly disable level 2 narrow metrics on r4:

```
[edit]
lab@r4# set protocols isis level 2 wide-metrics-only

[edit]
lab@r4# commit
commit complete
```

After a few moments, we finally observe the required load-balancing behavior on r5:

```
[edit]
lab@r5# run show route 10.0.5/24

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.5.0/24      *[IS-IS/18] 00:00:20, metric 520, tag 2
                 > to 10.0.2.2 via at-0/2/1.35
                 <u>to 10.0.2.10 via as1.0</u>
```

To complete this configuration example, connectivity to the 10.0.5/24 prefix is tested from the perspective of a level 1 router in area 49.0002:

```
lab@r7> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 10.0.8.9 (10.0.8.9) 0.338 ms 0.242 ms 0.216 ms
 2 10.0.2.2 (10.0.2.2) 0.777 ms 0.656 ms 0.878 ms
 3 <u>10.0.4.2 (10.0.4.2) 0.565 ms 0.655 ms 0.379 ms</u>
 4 10.0.5.1 (10.0.5.1) 0.690 ms 0.688 ms 0.878 ms
```

```
lab@r7> traceroute 10.0.5.2
traceroute to 10.0.5.2 (10.0.5.2), 30 hops max, 40 byte packets
 1 10.0.8.9 (10.0.8.9) 0.342 ms 0.241 ms 0.216 ms
 2 10.0.2.2 (10.0.2.2) 0.863 ms 1.162 ms 0.378 ms
 3 10.0.5.2 (10.0.5.2) 0.276 ms 0.643 ms 0.
```

The (highlighted) extra hop through r2 to reach the 10.0.5.1 address could be the normal result of IS-IS load balancing from r3 to r1 and r2, if we assume that r2 has been chosen as the current next hop for the 10.0.5/24 subnet. The following capture confirms this suspicion and proves that all is well with the route redistribution and redundancy requirements of your configuration:

```
lab@r3> show route 10.0.5.1

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.5.0/24          *[IS-IS/15] 00:12:16, metric 510, tag 1
                    to 10.0.4.14 via fe-0/0/0.0
                    > to 10.0.4.2 via fe-0/0/1.0
```

Congratulations! You have met the requirements specified for the 10.0.5/24 prefix, with regard to its redistribution, metric setting, and backbone load-balancing capabilities.

While not required for this example, it is suggested that you now disable narrow metrics (at level 2) on r5, and at level 1 on r3 and r4. Mixing narrow and wide metrics within a single area can cause confusion down the road.

IS-IS and OSPF Redistribution on r6

You will now configure r6 to redistribute the default route from IS-IS into OSPF, and in the reverse direction, to redistribute the 192.168.x/24 OSPF routes into IS-IS. We begin with the following command that configures r6 to run OSPF on its fe-0/1/3 interface in area 1:

```
[edit protocols]
lab@r6# set ospf area 1 interface fe-0/1/3
```

After committing the changes, we verify that routes are now being received from the OSPF router:

```
[edit protocols]
lab@r6# run show route protocol ospf

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```

192.168.0.0/24    *[OSPF/150] 00:00:16, metric 0, tag 0
                 > to 172.16.40.1 via fe-0/1/3.0
192.168.1.0/24    *[OSPF/150] 00:00:16, metric 0, tag 0
                 > to 172.16.40.1 via fe-0/1/3.0
192.168.2.0/24    *[OSPF/150] 00:00:16, metric 0, tag 0
                 > to 172.16.40.1 via fe-0/1/3.0
192.168.3.0/24    *[OSPF/150] 00:00:16, metric 0, tag 0
                 > to 172.16.40.1 via fe-0/1/3.0
224.0.0.5/32     *[OSPF/10] 00:01:23, metric 1

```

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

Great, r6 has formed an adjacency to the OSPF router. The next step is to write an export policy to redistribute the OSPF routes into IS-IS:

```

[edit]
lab@r6# edit policy-options policy-statement ospf-isis

[edit policy-options policy-statement ospf-isis]
lab@r6# set term 1 from protocol ospf

[edit policy-options policy-statement ospf-isis]
lab@r6# set term 1 from route-filter 192.168.0/22 longer

[edit policy-options policy-statement ospf-isis]
lab@r6# set term 1 then accept

```

The resulting OSPF to IS-IS redistribution policy is shown next:

```

[edit policy-options policy-statement ospf-isis]
lab@r6# show
term 1 {
  from {
    protocol ospf;
    route-filter 192.168.0.0/22 longer;
  }
  then accept;
}

```

The *ospf-isis* policy is now applied to the IS-IS instance as an export policy:

```

[edit]
lab@r6# set protocols isis export ospf-isis

```

With the export policy in place on r6 and the changes committed, we confirm the presence of the 192.168.x/24 routes on r5 to determine its effect:

```
[edit]
lab@r5# run show route 192.168.0/22

inet.0: 26 destinations, 26 routes (26 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.0/24    *[IS-IS/160] 00:00:01, metric 10, tag 1
                  > to 10.0.8.5 via fe-0/0/0.0
192.168.1.0/24    *[IS-IS/160] 00:00:01, metric 10, tag 1
                  > to 10.0.8.5 via fe-0/0/0.0
192.168.2.0/24    *[IS-IS/160] 00:00:01, metric 10, tag 1
                  > to 10.0.8.5 via fe-0/0/0.0
192.168.3.0/24    *[IS-IS/160] 00:00:01, metric 10, tag 1
                  > to 10.0.8.5 via fe-0/0/0.0
```

So far, so good—all the OSPF routes are present on r5 as IS-IS level 1 external routes. To get the routes into the IS-IS backbone, you must now write an export policy on r5 instructing it to redistribute level 1 externals into level 2. The *l1-ext* policy shown here works well for our purposes:

```
[edit policy-options policy-statement l1-ext]
lab@r5# show
term 1 {
  from {
    protocol isis;
    level 1;
    route-filter 192.168.0.0/22 longer;
  }
  to level 2;
  then accept;
}
```

After applying the *l1-ext* as export to r5's IS-IS instance, the presence of the 192.168.x routes are confirmed on r3:

```
lab@r3> show route 192/8

inet.0: 33 destinations, 33 routes (33 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```

192.168.0.0/24    *[IS-IS/18] 00:00:01, metric 20, tag 2
                  > to 10.0.2.1 via at-0/1/0.35
192.168.0.1/32  *[IS-IS/18] 00:00:01, metric 21, tag 2
                  > to 10.0.2.1 via at-0/1/0.35
192.168.1.0/24  *[IS-IS/18] 00:00:01, metric 20, tag 2
                  > to 10.0.2.1 via at-0/1/0.35
192.168.2.0/24  *[IS-IS/18] 00:00:01, metric 20, tag 2
                  > to 10.0.2.1 via at-0/1/0.35
192.168.3.0/24  *[IS-IS/18] 00:00:01, metric 20, tag 2
                  > to 10.0.2.1 via at-0/1/0.35

```

In this example, r3 displays a preference of 18 for the 192.168.0/22 routes because the use of level 2 wide metrics on r5 prevents r3 from seeing the routes as level 2 externals, which have a default preference of 165. Refer back to the previous section, “Troubleshoot IS-IS Redistribution,” for details on why wide metrics have been enabled in the level 2 backbone. With OSPF-to-IS-IS redistribution working properly at r6 and r5, you now create and apply the policy needed to redistribute the IS-IS default route from r6 into OSPF. The policy shown next will accommodate the requirements of this example, but other policy approaches are also workable.

```

[edit policy-options policy-statement default]
lab@r6# show
term 1 {
  from {
    route-filter 0.0.0.0/0 exact;
  }
  then accept;
}

```

Now apply the *default* policy as an export to OSPF using the following command:

```

[edit]
lab@r6# set protocols ospf export default

```

After committing the change, confirm that the default route is present on the OSPF router:

```

lab@ospf> show route protocol ospf
inet.0: 17 destinations, 17 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[OSPF/150] 00:02:47, metric 10, tag 1
                  > to 172.16.40.2 via fe-0/0/0.0
10.0.9.6/32       *[OSPF/10] 00:22:38, metric 1
                  > to 172.16.40.2 via fe-0/0/0.0
224.0.0.5/32      *[OSPF/10] 00:23:23, metric 1

```

The default route is present as required, but unfortunately so is the route to r6's loopback address. In accordance with the requirements of this example, you are to advertise *only* the default route to the OSPF router. Solving this problem can also be tricky. The issue relates to the JUNOS software behavior of automatically advertising a stub route to the interface from which the OSPF RID was obtained, as described in Chapter 3, "OSPF Configuration and Testing." To eliminate this route, you must manually assign r6's RID as shown:

```
[edit]
lab@r6# set routing-options router-id 10.0.9.6
```

```
[edit]
lab@r6# commit
commit complete
```

Once again, confirm the results from the perspective of the OSPF router:

```
lab@ospf> show route protocol ospf

inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[OSPF/150] 00:08:11, metric 10, tag 1
                   > to 172.16.40.2 via fe-0/0/0.0
224.0.0.5/32      *[OSPF/10] 00:28:47, metric 1
```

As required, r6 is now advertising only the default route to the OSPF router. With this 5.2 code base, a routing daemon restart was not required for the change in RID source to take effect. With some JUNOS software versions, you may have to restart routing (or for dramatic effect, reboot the router) in order to have the routing daemon begin using the new source of its RID. With the default route confirmed in the OSPF router, you decide to test connectivity to the 10.0.5/24 prefix before bringing r7 into the route redistribution fray:

```
lab@ospf> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 172.16.40.2 (172.16.40.2) 0.190 ms 0.123 ms 0.096 ms
 2 * * *
^C
```

This problem should seem familiar, as the very same issue arose in the OSPF-to-RIP redistribution example provided in Chapter 3. The issue is the lack of a route in the IS-IS routing domain for the 172.16.40/24 OSPF subnet used by the OSPF router when sourcing its packets. This is confirmed by using the source switch to the traceroute command as shown next:

```
lab@ospf> traceroute 10.0.5.1 source 192.168.0.1
traceroute to 10.0.5.1 (10.0.5.1) from 192.168.0.1, 30 hops max, 40 byte packets
```

```

1 172.16.40.2 (172.16.40.2) 0.190 ms 0.122 ms 0.094 ms
2 10.0.8.6 (10.0.8.6) 0.321 ms 0.289 ms 0.277 ms
3 10.0.2.2 (10.0.2.2) 0.756 ms 0.705 ms 0.874 ms
4 10.0.4.2 (10.0.4.2) 0.621 ms 0.673 ms 0.374 ms
5 10.0.5.1 (10.0.5.1) 0.628 ms 0.704 ms 0.875 ms

```

Because no restrictions have been placed on how you should handle this problem (other than your not being able to modify the OSPF router's configuration), you could opt for one of several solutions that include running IS-IS passive on the OSPF subnet, or the redistribution of the direct interface route from r6. Because the latter approach will also require a policy modification on r5 to tell it that it should also redistribute the 172.16.40/24 level 1 external into the backbone, this author has opted to go with the simpler passive IGP approach, as shown next:

```

[edit protocols]
lab@r6# set isis interface fe-0/1/3 level 2 disable

```

```

[edit protocols]
lab@r6# set isis interface fe-0/1/3 passive

```

```

[edit interfaces]
lab@r6# set fe-0/1/3 unit 0 family iso

```

After committing changes on r6, we again test reachability to 10.0.5/24 from the OSPF router:

```

lab@ospf> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
1 172.16.40.2 (172.16.40.2) 0.195 ms 0.123 ms 0.094 ms
2 10.0.8.6 (10.0.8.6) 0.341 ms 0.304 ms 0.278 ms
3 10.0.2.2 (10.0.2.2) 0.777 ms 1.204 ms 0.876 ms
4 10.0.4.2 (10.0.4.2) 0.400 ms 0.682 ms 0.375 ms
5 10.0.5.1 (10.0.5.1) 0.845 ms 1.204 ms 0.872 ms

```

IS-IS and OSPF Redistribution on r7

With IS-IS to OSPF route redistribution working on r6, you must now configure r7 so that it too redistributes routes between IS-IS and OSPF. As always, the bidirectional redistribution of routes from multiple locations is a task fraught with the peril of a routing loop. Before attacking this problem, it might behoove you to consider that the default preference for the IS-IS default route is 15, while the OSPF external route that results from its redistribution into OSPF will have a preference of 150. This means that r6 and r7 will automatically “ignore” the redistributed default route should it later be received over the OSPF subnet. Further, if this default should be inadvertently re-advertised back into the IS-IS level 1 area, the level 1 external preference will be 160, which will result in the continued use of the default route being injected by r5, which is a level 1 internal with a preference of 15. The same preference settings should also cause r6

and r7 to make the “right” choice when presented with the 192.168.x routes as both OSPF routes and IS-IS level 1 external routes, which is to say that the routers will prefer the OSPF source as desired.

Based on the default route preference settings, it would seem that a loop will not occur if the somewhat sloppy export policy in use by r6 is simply copied over to r7, even though such a procedure did result in a routing loop when OSPF was used to redistribute routes to and from the RIP protocol in Chapter 3. To put the theory to the test, r6’s export policies are copied and committed on r7. You should also configure r7’s OSPF area 1, and passive IS-IS interface as shown for r6 in the previous section. Lastly, do not forget to add the manually assigned RID to r7 to prevent it from advertising a stub route to its loopback address.

Verify OSPF and IS-IS Redistribution

When the configuration of r7 is complete, the results are confirmed with the following commands, starting with the verification that the OSPF router sees two equal-cost next hops for the default route:

```
lab@ospf> show route protocol ospf
```

```
inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
0.0.0.0/0          *[OSPF/150] 00:01:32, metric 10, tag 1
                  > to 172.16.40.2 via fe-0/0/0.0
                  to 172.16.40.3 via fe-0/0/0.0
224.0.0.5/32      *[OSPF/10] 01:49:48, metric 1
```

We now confirm that r6 and r7 prefer the OSPF 192.168.x/24 externals, as received from the OSPF router, to the redistributed IS-IS level 1 externals:

```
[edit]
```

```
lab@r7# run show route 192/8
```

```
inet.0: 20 destinations, 21 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
192.168.0.0/24    *[OSPF/150] 00:01:26, metric 0, tag 0
                  > to 172.16.40.1 via fe-0/3/3.0
192.168.0.1/32   *[OSPF/10] 00:01:26, metric 1
                  > to 172.16.40.1 via fe-0/3/3.0
192.168.1.0/24   *[OSPF/150] 00:01:26, metric 0, tag 0
                  > to 172.16.40.1 via fe-0/3/3.0
192.168.2.0/24   *[OSPF/150] 00:01:26, metric 0, tag 0
                  > to 172.16.40.1 via fe-0/3/3.0
```

312 Chapter 4 • IS-IS Configuration and Testing

```
192.168.3.0/24    *[OSPF/150] 00:01:26, metric 0, tag 0  
                > to 172.16.40.1 via fe-0/3/3.0
```

You should also verify that the IS-IS level 1 default route is preferred to the OSPF external default on r6 and r7:

```
[edit]  
lab@r6# run show route 3.3.3.3  
  
inet.0: 20 destinations, 21 routes (20 active, 0 holddown, 0 hidden)  
+ = Active Route, - = Last Active, * = Both  
  
0.0.0.0/0          *[IS-IS/15] 02:14:46, metric 10, tag 1  
                > to 10.0.8.6 via fe-0/1/0.0  
                [OSPF/150] 00:15:57, metric 10, tag 1  
                > to 172.16.40.3 via fe-0/1/3.0
```

These results indicate that, as predicted, the default route preference settings for IS-IS and OSPF internal/external routes result in the routers just “doing the right thing” in that no loops have been created by the mutual route redistribution actions of r6 and r7. The last step is to confirm that r5 sees two equal-cost next hops for the redistributed 192.168.x/24 OSPF routes, which verifies that both r6 and r7 are correctly exporting the OSPF routes into IS-IS:

```
[edit]  
lab@r5# run show route 192.168.0/24  
  
inet.0: 27 destinations, 27 routes (27 active, 0 holddown, 0 hidden)  
+ = Active Route, - = Last Active, * = Both  
  
192.168.0.0/24    *[IS-IS/160] 00:06:11, metric 10, tag 1  
                > to 10.0.8.5 via fe-0/0/0.0  
                > to 10.0.8.10 via fe-0/0/1.0  
192.168.0.1/32  *[IS-IS/160] 00:07:21, metric 11, tag 1  
                > to 10.0.8.5 via fe-0/0/0.0  
                > to 10.0.8.10 via fe-0/0/1.0
```

All indications point to a properly functioning, loop-free network, which will provide both optimal routing and the required redundancy. A few quick traceroutes provide the final confirmation that you have met the requirements of the route redistribution example:

```
lab@ospf> traceroute 10.0.5.1  
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets  
 1 172.16.40.2 (172.16.40.2) 0.189 ms 0.123 ms 0.094 ms  
 2 10.0.8.6 (10.0.8.6) 0.329 ms 0.288 ms 0.279 ms
```

```

3 10.0.2.2 (10.0.2.2) 1.029 ms 1.204 ms 0.876 ms
4 10.0.4.2 (10.0.4.2) 0.323 ms 0.677 ms 0.372 ms
5 10.0.5.1 (10.0.5.1) 0.924 ms 1.186 ms 0.874 ms

```

The OSPF router has the required connectivity through r6, so we quickly test that r7 can also correctly forward should it be chosen by the OSPF router as the forwarding next hop the next time a load-balancing hash is performed:

```

lab@r7> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 10.0.8.9 (10.0.8.9) 0.326 ms 0.242 ms 0.216 ms
 2 10.0.2.2 (10.0.2.2) 0.839 ms 0.657 ms 0.880 ms
 3 10.0.4.2 (10.0.4.2) 0.499 ms 0.653 ms 0.379 ms
 4 10.0.5.1 (10.0.5.1) 0.759 ms 0.697 ms 0.866 ms

```

The traceroutes produce the expected results, which brings this configuration example to a close. You will not need to source traffic from the 10.0.5/24 subnet when testing connectivity to the OSPF router from either r1 or r2 because this example provides a default route to the OSPF router, which affords it connectivity to the whole IS-IS routing domain.

Congratulations!

Miscellaneous IS-IS Knobs

You should be familiar with all IS-IS options and available knobs before attempting your lab examination, because the evolving nature of the test will make it hard to predict when you will need to enable what might be considered a corner-case knob. The following section will demonstrate how some of the remaining IS-IS configuration parameters may be deployed in a lab scenario based on Figure 4.3. Traffic Engineering (TE)-related OSPF options are beyond the scope of the JNCIP examination and are therefore not covered in this book.

To complete this section, you must perform the following tasks:

- Ensure that r2 cannot reach destinations outside of area 49.0003 while keeping all its IS-IS adjacencies up and without modifying its `routing-options` stanza.
- Configure r6 with a level 1 priority of 0.
- Ensure that r5 never generates a non-0 selector byte LSP for the 10.0.8.8/30 network.
- Configure the r3-r5 ATM link so that flooded LSPs are spaced at least 300 milliseconds apart.

The first configuration task requires that we determine how r2 is currently forwarding traffic into other areas:

```

lab@r2> show route 10.0.2.0

inet.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)

```

+ = Active Route, - = Last Active, * = Both

```
0.0.0.0/0          *[IS-IS/15] 03:21:43, metric 10, tag 1
                   > to 10.0.4.9 via fe-0/0/1.0
                   to 10.0.4.1 via fe-0/0/2.0
```

As expected, r2 is using the default route produced by detecting the presence of a L1/L2-attached router to route to destinations outside of its level 1 area. There may be times, such as during a Denial of Service (DoS) attack, that you will not want a level 1 router to be able to forward traffic based on a default route. To prevent r2 from being able to reach inter-area destinations in accordance with the provided criteria, you must stop r2 from installing this default route without affecting the status of its IS-IS adjacencies. The following command is used to tell the router to ignore the presence of the attached bit in level 1 LSPs, which blocks the installation of the IS-IS default route:

```
[edit]
lab@r2# set protocols isis ignore-attached-bit
```

The results are confirmed by verifying that r2 no longer displays a default route for inter-area destinations:

```
lab@r2> show route 10.0.2.0
```

```
lab@r2>
```

The second and third configuration tasks relate to one another and are made difficult by the somewhat cryptic wording of the requirements. A reader who possesses a detailed understanding of the IS-IS protocol will realize that only an IS-IS DIS can generate non-0 selector byte LSPs as part of its pseudonode functionality. So to meet this requirement, you must ensure that r5 can never be elected as the IS-IS DIS for the 10.0.8.8/30 subnet, despite the need to set r6's DIS priority to 0!

This seemingly insurmountable task can only be accomplished by setting r5's priority to 0 and then manually assigning a MAC address to r5's fe-0/0/0 interface that is numerically lower than the MAC address in use by r6's fe-0/1/0 interface (or vice versa). Simply modifying r5's fe-0/0/0 interface priority will not suffice for this task, because the resulting priority tie with r6 will simply be resolved based on their burned-in MAC addresses. Though you may get lucky and find that r5 happens to lose the MAC address-based DIS election in your test bed, you would likely still lose points on the actual JNCIP exam for not taking the necessary precautions that would have allowed you to forsake the services of that fleeting mistress known as luck.

The following commands set the level 1 priority of r5's fe-0/0/0 interface to 0, and assigns a new MAC address that is numerically lower than r6's fe-0/1/0 MAC address, which in this test bed happens to be 00:90:69:32:11:3a:

```
[edit protocols isis]
lab@r5# set interface fe-0/0/0 level 1 priority 0
```

```
[edit interfaces fe-0/0/0]
lab@r5# set mac 00.00.00.00.00.11
```

The level 1 priority of r6 is now set to 0 with the following command. You can opt for a global change, as demonstrated here, or the explicit setting of the level 1 priority associated with its fe-0/1/0 interface:

```
[edit]
lab@r6# set protocols isis interface all level 1 priority 0
```

To confirm the expected behavior, we verify that r5 has not been elected the DIS on its fe-0/0/0 interface. Alternatively, we can examine the level 1 area's database to confirm that r5 has not generated a pseudonode LSP for the 10.0.8.4/30 subnet. Both approaches are shown here:

```
[edit]
lab@r5# run show isis interface fe-0/0/0
IS-IS interface database:
Interface          L CirID Level 1 DR          Level 2 DR          L1/L2 Metric
fe-0/0/0.0         1  0x2  r6.02                    Disabled            5/5
```

As required, r6 has won the DIS election, despite its priority setting of zero.

```
[edit]
lab@r5# run show isis database r5 detail
IS-IS level 1 link-state database:

r5.00-00 Sequence: 0x1b, Checksum: 0x63a3, Lifetime: 3597 secs
  IS neighbor:                r5.03 Metric:        5
  IS neighbor:                r6.02 Metric:        5
  IP prefix:                   10.0.2.0/23 Metric:   10 External
  IP prefix:                   10.0.8.8/30 Metric:   5 Internal
  IP prefix:                   10.0.8.4/30 Metric:   5 Internal

r5.03-00 Sequence: 0x9, Checksum: 0x1db, Lifetime: 3597 secs
  IS neighbor:                r7.00 Metric:        0
  IS neighbor:                r5.00 Metric:        0
```

```
IS-IS level 2 link-state database:
```

```
. . .
```

As expected, the only level 1, non-zero selector byte LSP generated by r5 refers to the 10.0.8.8/30 subnet by virtue of the neighbor report that includes r5 and r7. These results confirm that the requirements of this configuration task have been met.



You may want to flush the IS-IS LSDB with the `clear isis database` command before you conduct your LSP analysis to ensure that stale LSPs, which can serve to add confusion, are no longer present.

The last requirement in this section is to configure the ATM interface between r3 and r5 so that flooded LSPs are spaced by at least 300 milliseconds. Controlling the rate of LSP flooding can be desirable on slow links, or when the device at the other end has processing limitations, because LSPs that are spaced too closely may be lost in such situations. The following command, entered on both r3 and r5, does the trick:

```
[edit protocols isis]
lab@r5# set interface at-0/2/1.35 lsp-interval 301
```

To confirm, we examine the interface's IS-IS parameters, being sure to compare the ATM interface's settings to the default LSP interval in use by its other point-to-point interface:

```
lab@r5> show isis interface detail
IS-IS interface database:
as1.0
  Index: 13, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: disabled
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    2          1      64    10      9      27
at-0/2/1.35
  Index: 9, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 301 ms, CSNP interval: disabled
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    2          1      64    10      9      27
. . .
```



Real World Scenario

A Monkey in Your Wrench?

A JNCIP candidate may be expected to demonstrate the ability to reverse engineer the configuration of an attached device by virtue of not being given the required details of its configuration. By way of example, consider the case where you must establish an adjacency to the OSPF router when its configuration is not made available to you. In such a situation, you will need to deploy various diagnostic utilities, such as protocol tracing or traffic monitoring, to determine the OSPF parameters that must be added to r6 and r7 for compatibility with the OSPF router (as shown earlier in Figure 4.3). The following example shows a candidate starting with a “bare bones” OSPF configuration that is intended only to enable OSPF protocol tracing:

```
[edit protocols ospf]
lab@r6# show
traceoptions {
  file test;
  flag hello detail;
  flag error detail;
}
area 0.0.0.0 {
  interface fe-0/1/3.0;
}
```

The results obtained from the trace file make it clear that the OSPF router is in area 1 and that authentication is not being used. The various time-stamps and tracing output also make the detection of the 8-second hello interval setting possible:

```
May 18 20:58:29 OSPF packet ignored: area mismatch (0.0.0.1) from 172.16.40.1
May 18 20:58:29 OSPF rcvd Hello 172.16.40.1 -> 224.0.0.5 (fe-0/1/3.0)
May 18 20:58:29 Version 2, length 44, ID 192.168.0.1, area 0.0.0.1
May 18 20:58:29 checksum 0x676d, authtype 0
May 18 20:58:29 mask 255.255.255.0, hello_ivl 8, opts 0x2, prio 128
May 18 20:58:29 dead_ivl 32, DR 172.16.40.1, BDR 0.0.0.0
May 18 20:58:32 OSPF sent Hello 172.16.40.2 -> 224.0.0.5 (fe-0/1/3.0)
May 18 20:58:32 Version 2, length 44, ID 10.0.9.6, area 0.0.0.0
May 18 20:58:32 checksum 0x1507, authtype 0
May 18 20:58:32 mask 255.255.255.0, hello_ivl 10, opts 0x2, prio 128
```

```
May 18 20:58:32 dead_ivl 40, DR 172.16.40.2, BDR 0.0.0.0
May 18 20:58:36 OSPF packet ignored: area mismatch (0.0.0.1) from 172.16.40.1
May 18 20:58:36 OSPF rcvd Hello 172.16.40.1 -> 224.0.0.5 (fe-0/1/3.0)
May 18 20:58:36 Version 2, length 44, ID 192.168.0.1, area 0.0.0.1
May 18 20:58:36 checksum 0x676d, authtype 0
May 18 20:58:36 mask 255.255.255.0, hello_ivl 8, opts 0x2, prio 128
May 18 20:58:36 dead_ivl 32, DR 172.16.40.1, BDR 0.0.0.0
```

Summary

This chapter provided numerous examples of IS-IS configuration tasks that are indicative of those you may encounter while taking your JNCIP lab examination. You should now feel comfortable with the commands needed to add the `iso` family to your router interfaces, and with the aspects of IS-IS routing instance configuration needed to support either level 1, level 2, or a level 1/level 2 configuration. IS-IS authentication options were also demonstrated, as was the use of protocol tracing to diagnose operational problems caused by incorrect or incompatible configurations.

The use of JUNOS software policy to effect IS-IS route leaking, summarization, and route redistribution were demonstrated, along with the operational commands that can be used to determine the status of your IS-IS IGP. The following case study reviews critical IS-IS configuration tasks in a configuration scenario designed to simulate the JNCIP testing environment.

Case Study: IS-IS

The following case study simulates a typical JNCIP-level IS-IS configuration scenario. You should refer to the criteria listing and the information in Figure 4.4, the case study topology, in order to complete the IS-IS case study. It is assumed that you will be adding the IS-IS related configuration on top of the interface configuration that was left from the case study at the end of Chapter 2, “Interface Configuration and Testing.” You should therefore first remove any protocol, policy, family iso interface statements, and any static route-related configuration resulting from the examples given in this chapter from all your routers before beginning the case study. Ideally you will start the case study by reloading your Chapter 2 baseline configuration using `load override`.

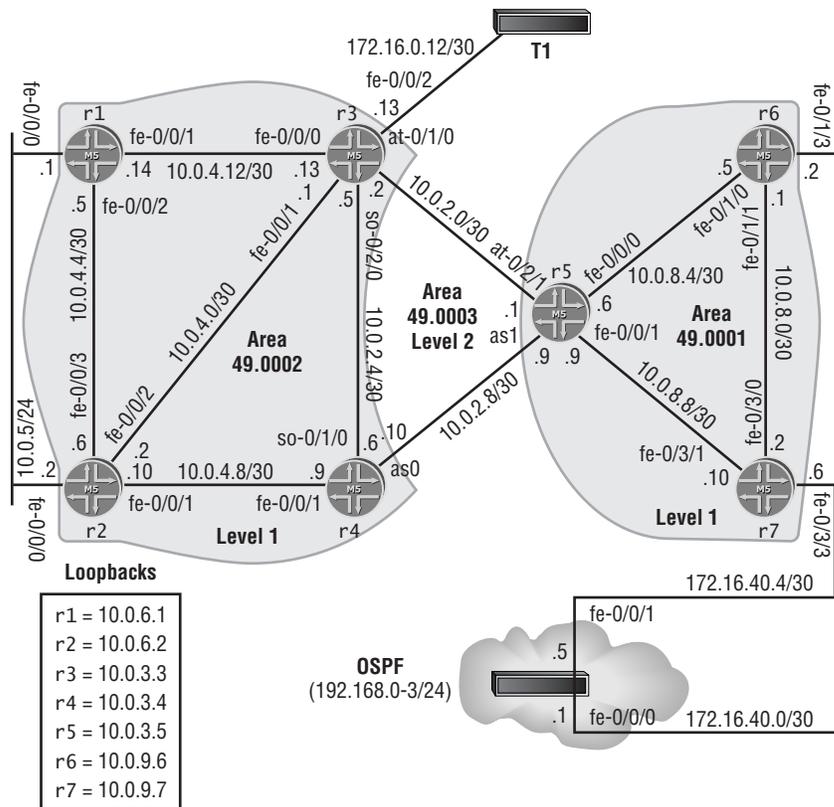
It is expected that a JNCIP candidate will be able to complete this case study in approximately one hour, with the result being an IS-IS IGP that exhibits no serious operational problems. Sample IS-IS configurations from all seven routers are provided at the end of the case study for comparison with your own configurations. Multiple solutions are sometimes possible, so differences between the provided examples and your own configurations do not always indicate that mistakes have been made. Because you are graded on the overall functionality of your IGP and its conformance to the specified configuration criteria, various operational mode commands are included so that you may compare the behavior of your network to a known good example.

To complete this case study, your IS-IS configuration must meet the following criteria:

- lo0 addresses are reachable through IS-IS for all routers with an IS-IS SysID based on the lo0 address. Backbone router lo0 addresses must not be injected into area 49.0002 or 49.0001.
- Backbone area routes must be summarized into area 49.0001.
- The subnet between r3 and T1 must appear in area 49.0003 as an internal route. Ensure that no adjacencies can be established on this subnet.
- Redistribute the 10.0.5/24 route into the backbone such that area 49.0002-attached routers see a metric of at least 100.
- r5 and r6 must use a level 1 priority of 0.
- Ensure that r7 never functions as a pseudonode when its adjacencies are up.
- Summarize all routes (internal and external) into the backbone area.
- You may not modify or view the OSPF router’s configuration.
- All areas must use hello authentication based on md5 with a secret of *jnx*. The backbone area must also authenticate LSPs using a simple password of *jnx*.
- Configure r6 and r7 to advertise the IS-IS default route to the OSPF router. Ensure that the OSPF router can load balance over the default route.
- Configure r6 and r7 to redistribute OSPF routes learned from the OSPF router into IS-IS.
- Except for r5, ensure that no single router or link failure will isolate r1, r2, or the OSPF router.

- Optimize routing based on bandwidth, and ensure that all Fast Ethernet interfaces are automatically assigned a metric of 5.
- No static routes and no loops.
- Set the level 1 preference in area 49.0001 to 155.
- You must not have suboptimal routing between r6, r7, and the OSPF router.
- Configure the network to keep LSPs that are up to 3600 seconds old.
- Only one adjacency is permitted between any two routers.

FIGURE 4.4 IS-IS case study topology



In this case study, the OSPF router is preconfigured and its configuration cannot be modified or viewed. You may assume, however, that it is configured to run OSPF, and that it has the policy statements needed to redistribute the 192.168.0-3/24 static routes into OSPF should an adjacency be formed.

IS-IS Case Study Analysis

Each configuration requirement for the case study will now be matched to one or more valid router configurations and the commands that can be used to confirm that the router is operating within all specified case study guidelines. We begin with these four criteria, as they serve to establish baseline IS-IS operation in each area:

- lo0 addresses are reachable through IS-IS for all routers with an IS-IS SysID based on the lo0 address. Backbone router lo0 addresses must not be injected into area 49.0002 or 49.0001.
- Optimize routing based on bandwidth, and ensure that all Fast Ethernet interfaces are automatically assigned a metric of 5.
- Configure the network to keep LSPs that are up to 3600 seconds old.
- Only one adjacency is permitted between any two routers.

To get basic IS-IS connectivity going, you must add the `iso` family to the correct logical units on all IS-IS interfaces, assign the router's NET to its lo0 address, and correctly configure the interface to level associations so that no interfaces are running at both levels 1 and 2. The following captures show typical IS-IS configurations from key routers throughout the network, starting with r4:

```
[edit]
lab@r4# run show interfaces terse
Interface      Admin Link Proto Local          Remote
fe-0/0/0       up   down
fe-0/0/1       up   up
fe-0/0/1.0     up   up   inet  10.0.4.9/30
-----
                               iso
. . .
so-0/1/0       up   up
so-0/1/0.100  up   up   inet  10.0.2.6/30
                               iso
so-0/1/1       up   up
so-0/1/1.0    up   up   soagg --> as0.0
so-0/1/2       up   up
so-0/1/2.0    up   up   soagg --> as0.0
so-0/1/3       up   down
as0            up   up
as0.0         up   up   inet  10.0.2.10/30
-----
                               iso
fxp0          up   up
fxp0.0        up   up   inet  10.0.1.4/24
fxp1          up   up
fxp1.0        up   up   tnp   4
```

```

. . .
lo0          up    up
lo0.0       up    up  inet  10.0.3.4          --> 0/0
                iso  49.0002.0100.0000.3004.00
. . .

```

The IS-IS routing instance for r4 is shown next:

```

[edit]
lab@r4# show protocols isis
reference-bandwidth 500m;
lsp-lifetime 3600;
interface fe-0/0/1.0 {
    level 2 disable;
}
interface all {
    level 1 disable;
}

```

r4's configuration has the required iso family interface assignments, NET, and the correct interface-to-level associations, which in this case was achieved by disabling level 1 on all interfaces with the exception of interface fe-0/0/1. Care must be taken to ensure that you do not configure backbone router lo0 interfaces to operate at level 1 in order to meet the requirement that backbone lo0 addresses are not to be injected into area 49.0002 or 49.0001. The automatic metric requirement is achieved by assigning reference bandwidth of 500Mbps, and the default LSP lifetime is set to 3600 seconds as required. Correct interface-to-level mapping and the automatically calculated metric values are confirmed by displaying IS-IS interface status:

```

[edit]
lab@r4# run show isis interface
IS-IS interface database:

```

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
as0.0	2	0x1	Disabled	Point to Point	<u>1/1</u>
fe-0/0/1.0	1	0x2	r2.02	Disabled	<u>5/5</u>
lo0.0	0	0x1	Disabled	Passive	0/0
so-0/1/0.100	2	0x1	Disabled	Point to Point	<u>3/3</u>

The configuration of routers in level 1 areas will be similar to the example shown next, which is taken from r7:

```

[edit]
lab@r7# run show interfaces terse
Interface          Admin Link Proto Local          Remote
. . .

```

```

fe-0/3/0      up    up
fe-0/3/0.0   up    up    inet  10.0.8.2/30
              iso
fe-0/3/1     up    up
fe-0/3/1.0   up    up    inet  10.0.8.10/30
              iso
. . .
lo0          up    up
lo0.0       up    up    inet  10.0.9.7          --> 0/0
              iso  49.0001.0100.0000.9007.00
. . .

```

The routing instance for a level 1 router r7 is configured as shown next:

```

[edit]
lab@r7# show protocols isis
reference-bandwidth 500m;
lsp-lifetime 3600;
interface all {
    level 2 disable;
}

```

The correct interface-to-level mappings are now confirmed on r7:

```

[edit]
lab@r7# run show isis interface
IS-IS interface database:

```

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
fe-0/3/0.0	1	0x2	r7.02	Disabled	5/5
fe-0/3/1.0	1	0x3	r5.03	Disabled	5/5
lo0.0	0	0x1	Passive	Disabled	0/0

At this stage, you should have the required number and type (either level 1 or level 2) of adjacencies between all seven routers. To quickly confirm, we verify that we have the expected number of SysIDs on a few strategic routers:

```

[edit]
lab@r3# run show isis hostname
IS-IS hostname database:

```

System Id	Hostname	Type
0100.0000.3003	r3	Static
0100.0000.3004	r4	Dynamic
0100.0000.3005	r5	Dynamic

324 Chapter 4 • IS-IS Configuration and Testing

```
0100.0000.6001 r1                               Dynamic
0100.0000.6002 r2                               Dynamic
```

r3 has hostname entries that correctly reflect area 49.0003 and 49.0002 routers, as expected. We confirm area 49.0001 by performing the same command on r6:

```
[edit]
lab@r6# run show isis hostname
IS-IS hostname database:
System Id      Hostname                                     Type
0100.0000.3005 r5                               Dynamic
0100.0000.9006 r6                               Static
0100.0000.9007 r7                               Dynamic
```

A test to confirm that backbone routers see IS-IS routes to all loopback addresses is now performed:

```
[edit protocols isis]
lab@r3# run show route protocol isis | match /32
10.0.3.4/32      *[IS-IS/18] 00:39:09, metric 3, tag 2
10.0.3.5/32      *[IS-IS/18] 00:39:09, metric 3, tag 2
10.0.6.1/32      *[IS-IS/15] 00:39:09, metric 5, tag 1
10.0.6.2/32      *[IS-IS/15] 00:39:09, metric 5, tag 1
10.0.9.6/32      *[IS-IS/18] 00:37:32, metric 8, tag 2
10.0.9.7/32      *[IS-IS/18] 00:36:27, metric 8, tag 2
```

As expected, there are six IS-IS routes on r3 that represent the loopback addresses of all remote routers. These results indicate that baseline IS-IS functionality is working as needed for the remaining configuration tasks. You will now add authentication as per this requirement:

- All areas must use hello authentication based on md5 with a secret of *jnx*. The backbone area must also authenticate LSPs using a simple password of *jnx*.

Backbone authentication is now added and confirmed. The authentication-related additions to a typical backbone router configuration are highlighted next:

```
[edit protocols isis]
lab@r5# show
reference-bandwidth 500m;
level 2 {
  authentication-key "$9$jEkmTOBEhrv"; # SECRET-DATA
  authentication-type simple; # SECRET-DATA
}
interface fe-0/0/0.0 {
  level 2 disable;
```

```

    level 1 {
        hello-authentication-key "$9$H.fz1IcSeW"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
interface fe-0/0/1.0 {
    level 2 disable;
    level 1 {
        hello-authentication-key "$9$EORS1M2gJZjq"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
interface all {
    level 1 disable;
    level 2 {
        hello-authentication-key "$9$5znCyrvmX-"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
}

```

Backbone authentication is confirmed by verifying the status of adjacencies (hello authentication) and the presence of IS-IS routes (LSP authentication):

```

[edit protocols isis]
lab@r4# run show isis adjacency

```

Interface	System	L State	Hold (secs)	SNPA
as0.0	r5	2 Up	20	
fe-0/0/1.0	0100.0000.6002	1 Down	24	0:a0:c9:b2:f8:cb
so-0/1/0.100	r3	2 Up	25	

```

[edit protocols isis]
lab@r4# run show isis route | match /32

```

10.0.3.3/32	2	55	3 int	so-0/1/0.100	r3
10.0.3.5/32	2	55	1 int	as0.0	r5

A working authentication configuration for a typical level 1 router is shown here with authentication-related additions highlighted:

```

[edit protocols isis]
lab@r1# show
reference-bandwidth 500m;
interface all {

```

```

hello-authentication-key "$9$MxL7Vji.mT3"; # SECRET-DATA
hello-authentication-type md5; # SECRET-DATA
level 2 disable;
}

```

Hello authentication in level 1 areas can be confirmed by verifying the correct adjacency status:

```

[edit]
lab@r2# run show isis adjacency

```

Interface	System	L State	Hold (secs)	SNPA
fe-0/0/1.0	r4	1 Up	25	0:90:69:6b:30:1
fe-0/0/2.0	r3	1 Up	19	0:90:69:6d:98:1
fe-0/0/3.0	r1	1 Up	25	0:a0:c9:6f:7b:84

Results such as these indicate that hello authentication has been correctly configured in level 1 areas. We next examine a working configuration example for the following case study requirements:

- Backbone area routes must be summarized into area 49.0001.
- The subnet between r3 and T1 must appear in area 49.0003 as an internal route. Ensure that no adjacencies can be established on this subnet.

The summarization and leaking of routes into area 49.0001 requires the use of policy and the definition of an aggregate route, as shown next in the configuration of r5, which highlights the configuration additions needed for this task:

```

[edit]
lab@r5# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
}
aggregate {
    route 10.0.2.0/23;
}

```

The policy needed to advertise an aggregate route into a level 1 area is shown here:

```

[edit]
lab@r5# show policy-options
policy-statement summ {
    term 1 {
        from {
            protocol aggregate;
            route-filter 10.0.2.0/23 exact;
        }
    }
}

```

```

        to level 1;
        then accept;
    }
}

```

The export policy must be applied to IS-IS as *export* for it to have any effect:

```

[edit]
lab@r5# show protocols isis export
export summ;

```

The correct operation of route summarization and leaking is easily confirmed on area 49.0001 routers:

```

[edit]
lab@r6# run show route 10.0.2/23

inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.2.0/23      *[IS-IS/160] 00:02:56, metric 15, tag 1
                 > to 10.0.8.6 via fe-0/1/0.0

```

Setting r3's fe-0/0/2 interface to *passive* ensures the 172.16.0.12/30 prefix is advertised into the backbone area as an internal route while also guarding against IS-IS adjacency formation:

```

[edit protocols isis]
lab@r3# show interface fe-0/0/2
passive;
level 1 disable;

```

It is a good idea to explicitly disable level 1 operation on this interface because its specific mention in the configuration results in its not being affected by the presence of the `interface all level 1 disable` clause. To confirm proper behavior, we verify that the route to 172.16.0.12/30 is present in the backbone as an IS-IS internal level 2 route:

```

[edit]
lab@r5# run show route 172/8

inet.0: 26 destinations, 26 routes (26 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.12/30  *[IS-IS/18] 02:10:36, metric 8, tag 2
                 > to 10.0.2.2 via at-0/2/1.35

```

Correct operation can also be verified by examining r3's level 2 LSP:

```
[edit]
lab@r5# run show isis database r3 detail
IS-IS level 1 link-state database:

IS-IS level 2 link-state database:

r3.00-00 Sequence: 0x2b, Checksum: 0x475a, Lifetime: 3551 secs. . .

IP prefix:                10.0.2.0/30 Metric:      3 Internal
IP prefix:                172.16.0.12/30 Metric:     5 Internal
-----
IP prefix:                10.0.4.0/30 Metric:      5 Internal
IP prefix:                10.0.4.12/30 Metric:      . . .
```

These results confirm that backbone routers are correctly seeing the 172.16.0.12/30 prefix as an internal level 2 IS-IS route, in accordance with the case study criteria. The correct LSP's lifetime, currently at 3551 seconds, is also highlighted here. We now address the following criteria:

- Redistribute the 10.0.5/24 route into the backbone while ensuring that area 49.0002-attached routers see a metric of at least 100.
- Except for r5, ensure that no single router or link failure will isolate r1, r2, or the OSPF router.

Achieving these requirements will involve the use of wide metrics for all level 1 area 49.0002 routers (including r3 and r4), and a route redistribution policy on both r1 and r2. Both routers must redistribute the prefix to ensure that a router failure will not sever communications between the OSPF router and the 10.0.5/24 prefix. Keep in mind that while this example is taken from r1, r2 will require similar modifications:

```
[edit]
lab@r1# show protocols isis
export direct;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 1 wide-metrics-only;
interface all {
    hello-authentication-key "$9$MexL7Vji.mT3"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
}
```

```
[edit]
lab@r1# show policy-options
```

```

policy-statement direct {
  term 1 {
    from {
      protocol direct;
      route-filter 10.0.5.0/24 exact;
    }
    then {
      metric 101;
      accept;
    }
  }
}

```

The results are confirmed on r3, where we expect to see both r1 and r2 as viable next hops and a metric greater than 100 for the 10.0.5/24 route:

```
[edit protocols isis]
```

```
lab@r3# run show route 10.0.5/24
```

```
inet.0: 27 destinations, 27 routes (27 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```

10.0.5.0/24      *[IS-IS/15] 00:01:34, metric 106, tag 1
                  to 10.0.4.14 via fe-0/0/0.0
                  > to 10.0.4.2 via fe-0/0/1.0

```

The use of wide metrics in area 49.0002 results in the automatic redistribution of the 10.0.5/24 prefix into the backbone area because the route will no longer be seen as an external prefix by the attached routers. Therefore, an IS-IS export policy is not required on r3 and r4 for this particular aspect of the case study. You must take care, however, to ensure that subsequent export policy applications to r3 and r4 correctly take this prefix into account. The following command confirms the route's presence in the backbone:

```
[edit]
```

```
lab@r5# run show route 10.0.5/24
```

```
inet.0: 37 destinations, 37 routes (37 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```

10.0.5.0/24      *[IS-IS/18] 00:01:21, metric 64, tag 2
                  > to 10.0.2.10 via as1.0

```

The use of narrow metrics, which support the route type attribute, would have necessitated a route-leaking policy on area 49.0002's attached routers to ensure the route injection into the

backbone area. The route's presence in r5, and the correct metric value displayed by area 49.0002 routers, confirms that the requirements for redistributing the 10.0.5/24 route have been met. Next, we will examine the configuration needed to meet the following criteria:

- r5 and r6 must use a level 1 priority of 0.
- Ensure that r7 never functions as a pseudonode when its adjacencies are up.
- Set the level 1 preference in area 49.0001 to 155.

With r5 and r6 set to a level 1 priority of 0, you will need to ensure that r7's priority is also set to 0 and that its MAC addresses are numerically lower than those used by r5 or r6 to prevent it from being elected a DIS. The following configuration prevents r7 from becoming the DIS, and correctly sets the IS-IS level 1 internal route preference in accordance with the case study requirements. The route preference and level 1 priority configuration is also needed on r5 and r6:

```
[edit]
lab@r7# show protocols isis
reference-bandwidth 500m;
lsp-lifetime 3600;
level 1 preference 155;
interface all {
    hello-authentication-key "$9$j5kmTOBEhrv"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
    level 1 priority 0;
}
```

The following highlights call out r7's new MAC addresses:

```
[edit]
lab@r7# show interfaces fe-0/3/0
mtu 4000;
mac 00.00.00.00.00.11;
unit 0 {
    family inet {
        address 10.0.8.2/30;
    }
    family iso;
}
```

```
[edit]
lab@r7# show interfaces fe-0/3/1
mac 00.00.00.00.00.22;
unit 0 {
```

```

family inet {
    address 10.0.8.10/30;
}
family iso;
}

```

The OSPF-to-IS-IS redistribution tasks must be completed before the level 1 preference modification can be confirmed. Viewing r7's IS-IS adjacencies and interface specifics to confirm the correct priority settings on r5, r6, and r7, and to verify that r7 is not functioning as a DIS, can be performed right away:

```

[edit]
lab@r7# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR          Level 2 DR          L1/L2 Metric
fe-0/3/0.0         1 0x2 r6.03                 Disabled            5/5
fe-0/3/1.0         1 0x3 r5.03                 Disabled            5/5
lo0.0              0 0x1 Passive                Disabled            0/0

```

These results confirm that r7 is currently not functioning as a DIS on any of its IS-IS interfaces. The following command allows you to confirm the required priority setting on r6 and r5:

```

[edit]
lab@r7# run show isis adjacency detail
r6
  Interface: fe-0/3/0.0, Level: 1, State: Up, Expires in 8 secs
  Priority: 0, Up/Down transitions: 3, Last transition: 00:08:03 ago
  Circuit type: 1, Speaks: IP, IPv6, MAC address: 0:a0:c9:69:c1:e4
  LAN id: r6.03, IP addresses: 10.0.8.1

r5
  Interface: fe-0/3/1.0, Level: 1, State: Up, Expires in 7 secs
  Priority: 0, Up/Down transitions: 5, Last transition: 00:08:04 ago
  Circuit type: 1, Speaks: IP, IPv6, MAC address: 0:90:69:69:70:1
  LAN id: r5.03, IP addresses: 10.0.8.9

```

The confirmation of the correct priority setting on r5 and r6, coupled with the knowledge that r7's level 1 interfaces have also had their MAC addresses manually set, confirms that the requirements for this configuration task have been met.

The following case study criteria will now be addressed:

- You may not modify or view the OSPF router's configuration.
- Configure r6 and r7 to advertise the IS-IS default route to the OSPF router. Ensure that the OSPF router can load balance over the default route.

- Configure r6 and r7 to redistribute OSPF routes learned from the OSPF router into IS-IS.
- Ensure that no single router or link failure will isolate r1, r2, or the OSPF router.
- You must not have suboptimal routing between r6, r7, and the OSPF router.

The OSPF-to-IS-IS redistribution task is made somewhat complex by the fact that you are not able to view or modify the OSPF router's configuration. To complete this task, you will need to perform a fair bit of reverse engineering, or develop some really good psychic skills! The following trace configuration will be used to get the ball rolling on r7, which should tell you something about this author's belief in ESP:

```
[edit]
lab@r7# show protocols ospf
traceoptions {
    file ospf;
    flag hello detail;
    flag error detail;
}
area 0.0.0.0 {
    interface fe-0/3/3.0;
}
```

After committing, monitor the trace file for clues to the OSPF router's configuration, which are highlighted next:

```
lab@r7# run monitor start ospf
```

```
[edit]
lab@r7#
*** ospf ***
May 22 23:31:22 OSPF sent Hello 172.16.40.6 -> 224.0.0.5 (fe-0/3/3.0)
May 22 23:31:22   Version 2, length 44, ID 10.0.9.7, area 0.0.0.0
May 22 23:31:22   checksum 0xe81c, authtype 0
May 22 23:31:22   mask 255.255.255.252, hello_ivl 10, opts 0x2, prio 128
May 22 23:31:22   dead_ivl 40, DR 0.0.0.0, BDR 0.0.0.0
May 22 23:31:23 OSPF packet ignored: area mismatch (0.0.0.2) from 172.16.40.5
May 22 23:31:23 OSPF rcvd Hello 172.16.40.5 -> 224.0.0.5 (fe-0/3/3.0)
May 22 23:31:23   Version 2, length 44, ID 192.168.0.1, area 0.0.0.2
May 22 23:31:23   checksum 0x6061, authtype 1
May 22 23:31:23   mask 255.255.255.252, hello_ivl 10, opts 0x8, prio 128
May 22 23:31:23   dead_ivl 40, DR 172.16.40.5, BDR 0.0.0.0
```

The highlighted entries reveal that the OSPF router is in area 2, and that it has been configured for simple password-based authentication (type 1). Though not overtly obvious, you can also

detect that the OSPF router is coding the options field with 0x08 while r7 is sending a 0x02. Knowing that the options field is used by OSPF to indicate area types and options, and knowing that r6 is set to area 0, which must be a transit area, you can assume that the OSPF router has been set to consider area 2 as some type of non-transit area such as a stub or NSSA. To obtain the correct authentication secret, you must perform traffic monitoring using the detail switch:

```
[edit]
lab@r7# run monitor traffic interface fe-0/3/3 detail
Listening on fe-0/3/3
23:40:57.778943 In 172.16.40.5 > 224.0.0.5: OSPFv2-hello 44: rtrid 192.168.0.1
area 0.0.0.2 auth "pass^@^@^@^@" mask 255.255.255.252 int 10 pri 128 dead 40 dr
172.16.40.5 nbrs [tos 0xc0] [ttl 1] (id 12499)
23:40:59.147114 Out 172.16.40.6 > 224.0.0.5: OSPFv2-hello 44: rtrid 10.0.9.7
backbone E mask 255.255.255.252 int 10 pri 128 dead 40 dr 172.16.40.6 nbrs [tos
0xc0] [ttl 1] (id 8628)
^C
2 packets received by filter
0 packets dropped by kernel
```

Armed with this information, we reconfigure r7 using the correct area and authentication parameters. In this case, we take a guess and decide to configure the area as a stub, based on the previously noted settings in the OSPF options field:

```
[edit protocols ospf]
lab@r7# delete area 0

[edit protocols ospf]
lab@r7# set area 2 authentication-type simple

[edit protocols ospf]
lab@r7# set area 2 interface fe-0/3/3

[edit protocols ospf]
lab@r7# set area 2 stub

[edit protocols ospf]
lab@r7# set area 2 interface fe-0/3/3 authentication-key pass
```

The modified configuration is displayed:

```
[edit protocols ospf]
lab@r7# show
traceoptions {
  file ospf;
```

```

    flag hello detail;
    flag error detail;
    flag packets detail;
}
area 0.0.0.2 {
    stub;
    authentication-type simple; # SECRET-DATA
    interface fe-0/3/3.0 {
        authentication-key "$9$6vcM/pBIRSeMXhS"; # SECRET-DATA
    }
}

```

After committing, you look for the required OSPF adjacency:

```

[edit protocols ospf]
lab@r7# run show ospf neighbor

```

```

[edit protocols ospf]
lab@r7#

```

No joy. OK, so perhaps instead of being a stub area, area 2 is set to operate as a NSSA? It's easy enough to find out by making the following change:

```

[edit protocols ospf]
lab@r7# set area 2 nssa

```

```

[edit protocols ospf]
lab@r7# commit
commit complete

```

The OSPF adjacency status is again checked:

```

[edit protocols ospf]
lab@r7# run show ospf neighbor

```

Address	Interface	State	ID	Pri	Dead
172.16.40.5	fe-0/3/3.0	<u>Full</u>	192.168.0.1	128	39

Great! The adjacency is up, which means r7 should now be receiving the routes that need to be redistributed into IS-IS. This policy statement matches on the 192.168.x/24 routes and advertises them into IS-IS when applied as export to IS-IS:

```

[edit]
lab@r6# show protocols isis export
export ospf-isis;

```

```
[edit]
lab@r6# show policy-options
policy-statement ospf-isis {
  term 1 {
    from {
      route-filter 192.168.0.0/22 longer;
    }
    then accept;
  }
  term 2 {
    from {
      route-filter 0.0.0.0/0 exact;
    }
    then reject;
  }
}
```

Term 2, which rejects the re-advertisement of the default route (in the case that it is coupled through the OSPF router), is not really needed because the default export policy will not redistribute OSPF routes into IS-IS. Still, it never hurts to be extra safe when dealing with route redistribution, so its inclusion is highly recommended. The OSPF export policy needed to redistribute the IS-IS default route to the OSPF router should look something like the following:

```
[edit]
lab@r6# show policy-options policy-statement isis-ospf
term 1 {
  from {
    route-filter 0.0.0.0/0 exact;
  }
  then accept;
}
```

After applying the *ospf-isis* and *isis-ospf* export policies to r6, the reason for requiring the change to the IS-IS level 1 preference in area 49.0001 becomes obvious. Changing the IS-IS level 1 preference as directed will cause r6 or r7 to prefer the default route being re-advertised by the OSPF router, which in turn will result in an extra hop when forwarding packets to the 10.0.5/24 subnet from either r6 or r7. Such a situation is shown next, where we see that r7 has chosen the OSPF default route over the IS-IS default route. At the time of this capture, both r6 and r7 have been configured to redistribute the IS-IS default into OSPF.

In this example, r6 has chosen the IS-IS default as the active route, while r7 has been fooled into accepting the OSPF default that is coupled through the OSPF router due to the level 1 preference modification:

```
[edit]
lab@r6# run show route | match 0.0.0.0/0
0.0.0.0/0          *[IS-IS/155] 01:08:38, metric 5, tag 1

[edit]
lab@r7# run show route | match 0.0.0.0/0

inet.0: 21 destinations, 28 routes (21 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[OSPF/150] 00:01:50, metric 5, tag 1
                   > to 172.16.40.5 via fe-0/3/3.0
                   [IS-IS/155] 01:09:32, metric 5, tag 1
                   > to 10.0.8.9 via fe-0/3/1.0
```

Because r7 has chosen the OSPF default, the *isis-ospf* export policy will have no effect because the currently active default route was learned through OSPF and, as such, will not be redistributed back into OSPF. This situation would result in point loss on the JNCIP exam both for having suboptimal routing and for the fact that the OSPF router cannot load balance over the default route because only r6 will be listed as the default route's next hop, as shown next:

```
lab@ospf> show route

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[OSPF/150] 00:03:19, metric 5, tag 1
                   > to 172.16.40.2 via fe-0/0/0.0
```

Because you cannot apply import policy to block the reception of the default route from the OSPF router, your only real choice here is to adjust the OSPF AS external route preference so that r6 and r7 once again prefer the IS-IS default route, which has a preference of 155 in this example. It is worth noting that the 192.168.x/24 routes being redistributed into IS-IS as level 1 externals will not cause problems if the preference associated with OSPF external routes is numerically lower, and therefore more preferred, than the default IS-IS level 1 external preference value of 160. So the OSPF external preference on r6 and r7 should be set to a value that is higher than 155 but lower than 160.

Based on the modified IS-IS level 1 preference value, setting the OSPF external preference to 159 on both r6 and r7 allows us to keep the default IS-IS level 1 external preference:

```
[edit]
lab@r6# set protocols ospf external-preference 159
```

After committing the change, r6 and r7 once again prefer the IS-IS default route to the OSPF default route, so both routers correctly export the IS-IS default into OSPF. This produces the required load-balancing behavior at the OSPF router and eliminates the suboptimal routing issue:

```
lab@ospf> show route
```

```
inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
0.0.0.0/0          *[OSPF/150] 00:00:43, metric 5, tag 1
                  to 172.16.40.2 via fe-0/0/0.0
                  > to 172.16.40.6 via fe-0/0/1.0
```

The last issue with the OSPF redistribution task is the need to somehow advertise the 172.16.40.0/29 prefixes used by the OSPF domain into area 49.0001. This can be accomplished by running passive IS-IS on these interfaces, or by modifying the IS-IS export policy so that it also redistributes these routes. The latter approach is reflected in this modified IS-IS export policy, which uses the longer match type to catch both of the OSPF subnets in a single route filter term:

```
[edit policy-options policy-statement ospf-isis]
lab@r6# show
term 1 {
  from {
    route-filter 192.168.0.0/22 longer;
    route-filter 172.16.40.0/29 longer;
  }
  then accept;
}
term 2 {
  from {
    route-filter 0.0.0.0/0 exact;
  }
  then reject;
}
```

With the modified IS-IS export policy on both r6 and r7, we expect to see that r5 can load balance to the 192.168.x/24 routes, which indicates that both level 1 routers are correctly redistributing the OSPF routes into IS-IS:

```
[edit]
lab@r5# run show route 192.168.0/24

inet.0: 34 destinations, 34 routes (34 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.0/24    *[IS-IS/160] 00:00:03, metric 5, tag 1
    > to 10.0.8.5 via fe-0/0/0.0
    to 10.0.8.10 via fe-0/0/1.0
192.168.0.1/32  *[IS-IS/160] 00:00:03, metric 6, tag 1
    > to 10.0.8.5 via fe-0/0/0.0
    to 10.0.8.10 via fe-0/0/1.0
```

With the mutual route redistribution apparently working, traceroutes are performed to confirm the required OSPF router to 10.0.5/24 subnet reachability:

```
lab@ospf> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 172.16.40.6 (172.16.40.6) 0.211 ms 0.167 ms 0.113 ms
 2 10.0.8.9 (10.0.8.9) 0.333 ms 0.304 ms 0.279 ms
 3 *^C
```

Well, these results are certainly less than stellar. A quick look at r3 sheds some light on the nature of the problem:

```
lab@r3> show route 192/8
```

```
lab@r3>
```

The absence of the redistributed routes in the backbone results from the JUNOS software IS-IS implementation's default policy of not leaking level 1 external prefixes into the level 2 backbone. This is remedied with the highlighted policy modification made to r5's IS-IS export policy:

```
[edit policy-options policy-statement summ]
lab@r5# show
term 1 {
    from {
        protocol aggregate;
        route-filter 10.0.2.0/23 exact;
    }
}
```

```

    to level 1;
    then accept;
}
term 2 {
  from {
    protocol isis;
    level 1;
    external;
  }
  to level 2;
  then accept;
}

```

After the change is committed, the traceroute is retested:

```

lab@ospf> traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 172.16.40.6 (172.16.40.6) 0.191 ms 0.123 ms 0.095 ms
 2 10.0.8.9 (10.0.8.9) 0.351 ms 0.293 ms 0.277 ms
 3 10.0.2.10 (10.0.2.10) 0.413 ms 0.695 ms 0.381 ms
 4 10.0.4.10 (10.0.4.10) 0.362 ms 0.701 ms 0.383 ms
 5 10.0.5.1 (10.0.5.1) 0.363 ms 0.700 ms 0.379 ms

```

Since the OSPF router is currently forwarding through r7, it is wise to also confirm the forwarding path from r6:

```

[edit]
lab@r6# run traceroute 10.0.5.1
traceroute to 10.0.5.1 (10.0.5.1), 30 hops max, 40 byte packets
 1 10.0.8.6 (10.0.8.6) 0.360 ms 0.297 ms 0.233 ms
 2 10.0.2.10 (10.0.2.10) 0.528 ms 0.495 ms 0.358 ms
 3 10.0.4.10 (10.0.4.10) 0.357 ms 0.647 ms 0.372 ms
 4 10.0.5.1 (10.0.5.1) 0.362 ms 0.683 ms 0.373 ms

```

The results indicate that IS-IS-to-OSPF route redistribution is working in accordance with the specified criteria. Before you consider this task complete, you should verify forwarding paths between r6, r7, and the OSPF router:

```

[edit]
lab@r7# run traceroute 10.0.9.6
traceroute to 10.0.9.6 (10.0.9.6), 30 hops max, 40 byte packets
 1 172.16.40.5 (172.16.40.5) 0.269 ms 0.196 ms 0.112 ms
 2 10.0.9.6 (10.0.9.6) 0.198 ms 0.185 ms 0.157 ms

```

The extra hop through the OSPF router is not good. This situation exists because both r6 and r7 are advertising a stub route to the interface that is sourcing their RID, and this stub route is being coupled through the OSPF router with a better preference than the IS-IS route for the same prefix:

```
[edit]
lab@r7# run show route 10.0.9.6

inet.0: 21 destinations, 24 routes (21 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.9.6/32      *[OSPF/10] 00:08:42, metric 2
                 > to 172.16.40.5 via fe-0/3/3.0
                 [IS-IS/160] 00:36:16, metric 5, tag 1
                 > to 10.0.8.1 via fe-0/3/0.0
```

While this issue could be resolved with preference adjustments, the cleanest fix is to manually configure the RID on both r6 and r7 as shown next:

```
[edit]
lab@r6# set routing-options router-id 10.0.9.6
```

After the change, you confirm that forwarding is now optimal:

```
[edit]
lab@r7# run traceroute 10.0.9.6
traceroute to 10.0.9.6 (10.0.9.6), 30 hops max, 40 byte packets
 1 10.0.9.6 (10.0.9.6) 0.201 ms 0.133 ms 0.100 ms
```

The forwarding problems caused by the automatic route advertisement for the interface used to provide the OSPF router ID could have easily been missed. This example should emphasize the need for JNCIP candidates to constantly check and re-check their work because of the difficulties of actually being able to predict this type of behavior.

To complete this case study, the following task must now be addressed:

- Summarize all routes (internal and external) into the backbone area.

Summarization of level 1 routes will require the local definition of one or more aggregate routes, and a corresponding policy that will block specific prefixes while advertising the aggregates into level 2 only. The following changes to r3 and r4 address the summarization of area 49.0002's routes:

```
[edit]
lab@r3# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
```

```

}
aggregate {
  route 10.0.4.0/22;
}
[edit]
lab@r3# show policy-options
policy-statement summ {
  term 1 {
    from {
      route-filter 10.0.5.0/24 exact;
    }
    to level 2;
    then accept;
  }
  term 2 {
    from {
      protocol aggregate;
      route-filter 10.0.4.0/22 exact;
    }
    to level 2;
    then accept;
  }
  term 3 {
    from {
      route-filter 10.0.4.0/22 longer;
    }
    to level 2;
    then reject;
  }
}

```

In this policy, **term 3** negates the default IS-IS policy of exporting IS-IS level 1 internals to level 2, which serves to suppress the more specific routes associated with area 49.0002. The second term is carefully written to ensure that the aggregate for area 49.0002 is injected only into the backbone area. The ordering of terms is significant in this policy because **term 3** will match and subsequently reject the 10.0.5/24 route if it is processed before **term 1**. The *summ* policy must be applied to r3 and r4 as an IS-IS export before it will have any effect:

```

[edit]
lab@r3# show protocols isis export
export summ;

```

The results are verified by checking the backbone to confirm the presence of the aggregate and the absence of the more specific contributing routes, except the 10.0.5/24 prefix, which must be leaked into the backbone:

```
[edit]
lab@r5# run show route 10.0.4/22

inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.4.0/22      *[IS-IS/165] 00:04:29, metric 11, tag 2
                  > to 10.0.2.10 via as1.0
10.0.5.0/24      *[IS-IS/18] 00:04:40, metric 64, tag 2
                  > to 10.0.2.10 via as1.0
```

Load-balancing from r5 to area 49.0002 is not occurring because of the metric differences between r3's at-0/1/0 and r4's as0 interfaces, which are using metrics of 3 and 1, respectively. Load-balancing could be achieved by adjusting the metrics of these links, but this is not a requirement in this case study.

Summarizing the routes from area 49.0001 requires the same technique, but care must be taken to also include an aggregate for the 192.168.x/24 routes originating in the OSPF router because the summarization requirements apply to both internal and external IS-IS routes. The following capture highlights the changes needed on r5 to correctly summarize area 49.0001 routes:

```
lab@r5# show routing-options aggregate
route 10.0.2.0/23;
route 10.0.8.0/21;
route 192.168.0.0/22;

[edit]
lab@r5# show policy-options
policy-statement summ {
    term 1 {
        from {
            protocol aggregate;
            route-filter 10.0.2.0/23 exact;
        }
        to level 1;
        then accept;
    }
    term 2 {
        from {
```

```

        route-filter 10.0.8.0/21 longer;
        route-filter 192.168.0.0/22 longer;
    }
    to level 2;
    then reject;
}
term 3 {
    from {
        protocol aggregate;
        route-filter 10.0.8.0/21 exact;
        route-filter 192.168.0.0/22 exact;
    }
    to level 2;
    then accept;
}
}
}

```

The results are now confirmed on a backbone router:

```
lab@r3> show route 10.0.8.0
```

```

inet.0: 25 destinations, 25 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.8.0/21      *[IS-IS/165] 00:02:32, metric 13, tag 2
                 > to 10.0.2.1 via at-0/1/0.35

```

```
lab@r3> show route 192/8
```

```

inet.0: 25 destinations, 25 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.0/22  *[IS-IS/165] 00:02:36, metric 13, tag 2
                 > to 10.0.2.1 via at-0/1/0.35

```

These results confirm that summarization into the IS-IS backbone has been correctly configured. Before calling it quits, it is recommended that you once again verify the required connectivity between the 10.0.5/24 subnet and the OSPF router:

```
lab@ospf> traceroute 10.0.5.2
```

```

traceroute to 10.0.5.2 (10.0.5.2), 30 hops max, 40 byte packets
 1 172.16.40.6 (172.16.40.6) 0.207 ms 0.121 ms 0.093 ms

```

```

2 10.0.8.9 (10.0.8.9) 0.344 ms 0.290 ms 0.277 ms
3 *^C

```

This is a good example of why assumptions should never be made in the JNCIP lab! Though not necessarily by design, you are likely to find that the layered nature of the configuration tasks can result in the need to modify existing configurations, which in turn can lead to the “breaking” of previously confirmed network functionality. In this case, the modification of r5’s policy, which was intended to achieve the case study’s summarization requirements, failed to take into account the prefixes associated with the OSPF subnets, thereby making the routes unreachable for backbone routers. The highlighted changes to r5’s IS-IS export policy resolves the remaining issue:

```

[edit]
lab@r5# show routing-options aggregate
route 10.0.2.0/23;
route 10.0.8.0/21;
route 192.168.0.0/22;
route 172.16.40.0/29;

[edit]
lab@r5# show policy-options
policy-statement summ {
  term 1 {
    from {
      protocol aggregate;
      route-filter 10.0.2.0/23 exact;
    }
    to level 1;
    then accept;
  }
  term 2 {
    from {
      route-filter 10.0.8.0/21 longer;
    }
    to level 2;
    then reject;
  }
  term 3 {
    from {
      protocol aggregate;
      route-filter 10.0.8.0/21 exact;
      route-filter 192.168.0.0/22 exact;
      route-filter 172.16.40.0/29 exact;
    }
  }
}

```

```

    }
    to level 2;
    then accept;
  }
}

```

This author has opted to summarize the OSPF subnets because the case study requirements make it clear that both internal and external IS-IS routes are to be summarized, and it is better to be safe than sorry. With the modified policy in place on r5, connectivity to the 10.0.5/24 subnet is restored and the goals of this case study achieved:

```

lab@ospf> traceroute 10.0.5.2
traceroute to 10.0.5.2 (10.0.5.2), 30 hops max, 40 byte packets
 1 172.16.40.6 (172.16.40.6) 0.193 ms 0.122 ms 0.093 ms
 2 10.0.8.9 (10.0.8.9) 0.332 ms 0.291 ms 0.277 ms
 3 10.0.2.10 (10.0.2.10) 0.336 ms 0.308 ms 0.305 ms
 4 10.0.5.2 (10.0.5.2) 0.254 ms 0.232 ms 0.217 ms

```

IS-IS Case Study Configurations

The modified configuration stanzas needed to complete the IS-IS case study are provided in Listings 4.4 through 4.10 for all routers in the test bed, with the IS-IS related changes highlighted.

Listing 4.4: r1 IS-IS Related Configuration

```

[edit]
lab@r1# show interfaces
fe-0/0/1 {
  vlan-tagging;
  unit 200 {
    vlan-id 200;
    family inet {
      address 10.0.4.14/30;
    }
    family iso;
  }
}
fe-0/0/2 {
  fastether-options {
    source-filtering;
    source-address-filter {
      00:a0:c9:6f:70:0d;
    }
  }
}

```

```

    }
    unit 0 {
        family inet {
            address 10.0.4.5/30;
        }
        family iso;
    }
}
fxp0 {
    unit 0 {
        family inet {
            address 10.0.1.1/24;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.6.1/32;
        }
        family iso {
            address 49.0002.0100.0000.6001.00;
        }
    }
}

```

[edit]

lab@r1# **show policy-options**

policy-statement direct {

 term 1 {

 from {

 protocol direct;

 route-filter 10.0.5.0/24 exact;

 }

 then {

 metric 101;

 accept;

 }

 }

}

```
[edit]
lab@r1# show protocols isis
export direct;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 1 wide-metrics-only;
interface all {
    hello-authentication-key "$9$MexL7Vji.mT3"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
}
}
```

Listing 4.5: r2 IS-IS Related Configuration

```
[edit]
lab@r2# show interfaces
fe-0/0/1 {
    speed 100m;
    link-mode half-duplex;
    fastether-options {
        no-flow-control;
    }
    unit 0 {
        family inet {
            address 10.0.4.10/30;
        }
        family iso;
    }
}
fe-0/0/2 {
    unit 0 {
        family inet {
            address 10.0.4.2/30;
        }
        family iso;
    }
}
fe-0/0/3 {
    fastether-options {
        source-filtering;
        source-address-filter {
            00:a0:c9:6f:7b:84;
        }
    }
}
```

```

    }
  }
  unit 0 {
    family inet {
      address 10.0.4.6/30;
    }
    family iso;
  }
}
fxp0 {
  unit 0 {
    family inet {
      address 10.0.1.2/24;
    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 10.0.6.2/32;
    }
    family iso {
      address 49.0002.0100.0000.6002.00;
    }
  }
}

```

[edit]

lab@r2# **show policy-options**

```

policy-statement direct {
  term 1 {
  from {
  protocol direct;
  route-filter 10.0.5.0/24 exact;
  }
  then {
  metric 101;
  accept;
  }
}
}

```

```
[edit]
lab@r2# show protocols isis
export direct;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 1 wide-metrics-only;
interface all {
    hello-authentication-key "$9$TF6Ar1MWxd"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
}
}
```

Listing 4.6: r3 IS-IS Related Configuration

```
[edit]
lab@r3# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
}
aggregate {
    route 10.0.4.0/22;
}
}
```

```
[edit]
lab@r3# show interfaces
fe-0/0/0 {
    vlan-tagging;
    unit 0 {
        vlan-id 200;
        family inet {
            mtu 1496;
            address 10.0.4.13/30;
        }
        family iso;
    }
}
fe-0/0/1 {
    unit 0 {
        family inet {
            address 10.0.4.1/30;
        }
        family iso;
    }
}
```

```

    }
}
fe-0/0/2 {
    unit 0 {
        family inet {
            address 172.16.0.13/30;
        }
        family iso;
    }
}
at-0/1/0 {
    atm-options {
        vpi 0 maximum-vcs 17;
        vpi 5 maximum-vcs 36;
        ilmi;
    }
    unit 35 {
        vci 5.35;
        shaping {
            cbr 50m;
        }
        oam-period 10;
        oam-liveness {
            up-count 2;
            down-count 2;
        }
        family inet {
            address 10.0.2.2/30;
        }
        family iso;
    }
}
so-0/2/0 {
    hold-time up 30 down 30;
    encapsulation frame-relay;
    lmi {
        n392dte 2;
        n393dte 3;
        t391dte 15;
        lmi-type itu;
    }
}

```

```

}
unit 100 {
  dlci 100;
  family inet {
    address 10.0.2.5/30;
  }
  family iso;
}
}
fxp0 {
  unit 0 {
    family inet {
      address 10.0.1.3/24;
    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 10.0.3.3/32;
    }
    family iso {
      address 49.0002.0100.0000.3003.00;
    }
  }
}
}

```

[edit]

lab@r3# **show policy-options**

```

policy-statement summ {
  term 1 {
  from {
  route-filter 10.0.5.0/24 exact;
  }
  to level 2;
  then accept;
  }
  term 2 {
  from {
  protocol aggregate;

```

```

route-filter 10.0.4.0/22 exact;
}
to level 2;
then accept;
}
term 3 {
from {
route-filter 10.0.4.0/22 longer;
}
to level 2;
then reject;
}
}

[edit]
lab@r3# show protocols isis
export summ;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 2 {
authentication-key "$9$DCH.500REyK"; # SECRET-DATA
authentication-type simple; # SECRET-DATA
}
level 1 wide-metrics-only;
interface fe-0/0/0.0 {
level 2 disable;
level 1 {
hello-authentication-key "$9$qPT3REyrvL"; # SECRET-DATA
hello-authentication-type md5; # SECRET-DATA
}
}
interface fe-0/0/1.0 {
level 2 disable;
level 1 {
hello-authentication-key "$9$FzZb6CuKvLX-w"; # SECRET-DATA
hello-authentication-type md5; # SECRET-DATA
}
}
interface fe-0/0/2.0 {
passive;
}

```

```

    level 1 disable;
  }
  interface all {
    level 1 disable;
    level 2 {
      hello-authentication-key "$9$JeUi.At0BEy"; # SECRET-DATA
      hello-authentication-type md5; # SECRET-DATA
    }
  }
}

```

Listing 4.7: r4 IS-IS Related Configuration

```

[edit]
lab@r4# show routing-options
static {
  route 10.0.200.0/24 next-hop 10.0.1.102;
}
aggregate {
  route 10.0.4.0/22;
}

[edit]
lab@r4# show interfaces
fe-0/0/1 {
  speed 100m;
  link-mode half-duplex;
  fastether-options {
    no-flow-control;
  }
  unit 0 {
    family inet {
      address 10.0.4.9/30;
    }
    family iso;
  }
}
so-0/1/0 {
  dce;
  hold-time up 30 down 30;
  encapsulation frame-relay;
  lmi {
    n392dce 2;
  }
}

```

```

        n393dce 3;
        t392dce 25;
        lmi-type itu;
    }
    unit 100 {
        dlci 100;
        family inet {
            address 10.0.2.6/30;
        }
        family iso;
    }
}
so-0/1/1 {
    sonet-options {
        path-trace r4-jnx;
        aggregate as0;
    }
}
so-0/1/2 {
    sonet-options {
        path-trace r4-jnx;
        aggregate as0;
    }
}
as0 {
    aggregated-sonet-options {
        minimum-links 2;
        link-speed oc3;
    }
    unit 0 {
        family inet {
            address 10.0.2.10/30;
        }
        family iso;
    }
}
fxp0 {
    unit 0 {
        family inet {
            address 10.0.1.4/24;
        }
    }
}

```

```

    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 10.0.3.4/32;
    }
    family iso {
      address 49.0002.0100.0000.3004.00;
    }
  }
}

```

```

[edit]
lab@r4# show policy-options
policy-statement summ {
  term 1 {
    from {
      route-filter 10.0.5.0/24 exact;
    }
    to level 2;
    then accept;
  }
  term 2 {
    from {
      protocol aggregate;
      route-filter 10.0.4.0/22 exact;
    }
    to level 2;
    then accept;
  }
  term 3 {
    from {
      route-filter 10.0.4.0/22 longer;
    }
    to level 2;
    then reject;
  }
}

```

```

[edit]
lab@r4# show protocols isis
export summ;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 2 {
    authentication-key "$$E7eS1M2gJZjq"; # SECRET-DATA
    authentication-type simple; # SECRET-DATA
}
level 1 wide-metrics-only;
interface fe-0/0/1.0 {
    level 2 disable;
    level 1 {
        hello-authentication-key "$$jdkmTOBEhrv"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
interface all {
    level 1 disable;
    level 2 {
        hello-authentication-key "$$vsc8xdDjq.5F"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}

```

Listing 4.8: r5 IS-IS Related Configuration

```

[edit]
lab@r5# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
}
aggregate {
    route 10.0.2.0/23;
    route 10.0.8.0/21;
    route 192.168.0.0/22;
    route 172.16.40.0/29;
}

[edit]
lab@r5# show interfaces
fe-0/0/0 {

```

```
unit 0 {
    family inet {
        address 10.0.8.6/30;
    }
    family iso;
}
fe-0/0/1 {
    unit 0 {
        family inet {
            address 10.0.8.9/30;
        }
        family iso;
    }
}
so-0/1/0 {
    sonet-options {
        path-trace r5-jnx;
        aggregate as1;
    }
}
so-0/1/1 {
    sonet-options {
        path-trace r5-jnx;
        aggregate as1;
    }
}
at-0/2/1 {
    atm-options {
        vpi 0 maximum-vcs 17;
        vpi 5 maximum-vcs 36;
        ilmi;
    }
    unit 35 {
        vci 5.35;
        shaping {
            cbr 50m;
        }
        oam-period 10;
        oam-liveness {
```

```
        up-count 2;
        down-count 2;
    }
    family inet {
        address 10.0.2.1/30;
    }
    family iso;
}
as1 {
    aggregated-sonet-options {
        minimum-links 2;
        link-speed oc3;
    }
    unit 0 {
        family inet {
            address 10.0.2.9/30;
        }
        family iso;
    }
}
fxp0 {
    unit 0 {
        family inet {
            address 10.0.1.5/24;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.3.5/32;
        }
        family iso {
            address 49.0001.0100.0000.3005.00;
        }
    }
}
```

```

[edit]
lab@r5# show policy-options
policy-statement summ {
  term 1 {
    from {
      protocol aggregate;
      route-filter 10.0.2.0/23 exact;
    }
    to level 1;
    then accept;
  }
  term 2 {
    from {
      route-filter 10.0.8.0/21 longer;
    }
    to level 2;
    then reject;
  }
  term 3 {
    from {
      protocol aggregate;
      route-filter 10.0.8.0/21 exact;
      route-filter 192.168.0.0/22 exact;
      route-filter 172.16.40.0/29 exact;
    }
    to level 2;
    then accept;
  }
}

lab@r5# show protocols isis
export summ;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 2 {
  authentication-key "$9$jEkmTOBEhrv"; # SECRET-DATA
  authentication-type simple; # SECRET-DATA
}
level 1 preference 155;
interface fe-0/0/0.0 {

```

```

    level 2 disable;
    level 1 {
        hello-authentication-key "$9$H.fz1IcSeW"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
        priority 0;
    }
}
interface fe-0/0/1.0 {
    level 2 disable;
    level 1 {
        hello-authentication-key "$9$EORS1M2gJZjq"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
        priority 0;
    }
}
interface all {
    level 1 disable;
    level 2 {
        hello-authentication-key "$9$5znCyrVMX-"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}

```

Listing 4.9: r6 IS-IS Related Configuration

```

[edit]
lab@r6# show interfaces
fe-0/1/0 {
    unit 0 {
        family inet {
            address 10.0.8.5/30;
        }
        family iso;
    }
}
fe-0/1/1 {
    mtu 4000;
    unit 0 {
        family inet {
            address 10.0.8.1/30;
        }
        family iso;
    }
}

```

```

    }
}
fe-0/1/3 {
    unit 0 {
        family inet {
            address 172.16.40.2/30;
        }
    }
}
fxp0 {
    unit 0 {
        family inet {
            address 10.0.1.6/24;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.9.6/32;
        }
        family iso {
            address 49.0001.0100.0000.9006.00;
        }
    }
}
[edit]
lab@r6# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
}
router-id 10.0.9.6;

[edit]
lab@r6# show policy-options
policy-statement isis-ospf {
    term 1 {
        from {
            route-filter 0.0.0.0/0 exact;
        }
    }
}

```

```

_____ then accept;
_____ }
_____ }
policy-statement ospf-isis {
_____ term 1 {
_____ from {
_____ route-filter 192.168.0.0/22 longer;
_____ route-filter 172.16.40.0/29 longer;
_____ }
_____ then accept;
_____ }
_____ term 2 {
_____ from {
_____ route-filter 0.0.0.0/0 exact;
_____ }
_____ then reject;
_____ }
_____ }
[edit]
lab@r6# show protocols
isis {
_____ export ospf-isis;
_____ reference-bandwidth 500m;
_____ lsp-lifetime 3600;
_____ level 1 preference 155;
_____ interface all {
_____ hello-authentication-key "$9$NoVs4PfzF/t"; # SECRET-DATA
_____ hello-authentication-type md5; # SECRET-DATA
_____ level 2 disable;
_____ level 1 priority 0;
_____ }
_____ }
ospf {
_____ external-preference 159;
_____ export isis-ospf;
_____ area 0.0.0.2 {
_____ nssa;
_____ authentication-type simple; # SECRET-DATA
_____ interface fe-0/1/3.0 {
_____ authentication-key "$9$Bvn1c1KvLNVYWL"; # SECRET-DATA

```

```

_____}
_____]
]

```

Listing 4.10: r7 IS-IS Related Configuration

```

[edit]
lab@r7# show interfaces
fe-0/3/0 {
  mtu 4000;
  mac 00.00.00.00.00.11;
  unit 0 {
    family inet {
      address 10.0.8.2/30;
    }
    family iso;
  }
}
fe-0/3/1 {
  mac 00.00.00.00.00.22;
  unit 0 {
    family inet {
      address 10.0.8.10/30;
    }
    family iso;
  }
}
fe-0/3/3 {
  unit 0 {
    family inet {
      address 172.16.40.6/30;
    }
  }
}
fxp0 {
  unit 0 {
    family inet {
      address 10.0.1.7/24;
    }
  }
}
lo0 {

```

```

unit 0 {
    family inet {
        address 10.0.9.7/32;
    }
    family iso {
        address 49.0001.0100.0000.9007.00;
    }
}
[edit]
lab@r7# show routing-options
static {
    route 10.0.200.0/24 next-hop 10.0.1.102;
}
router-id 10.0.9.7;

[edit]
lab@r7# show policy-options
policy-statement isis-ospf {
    term 1 {
        from {
            route-filter 0.0.0.0/0 exact accept;
        }
    }
}
policy-statement ospf-isis {
    term 1 {
        from {
            route-filter 192.168.0.0/22 longer;
            route-filter 172.16.40.0/29 longer;
        }
        then accept;
    }
    term 2 {
        from {
            route-filter 0.0.0.0/0 exact;
        }
        then reject;
    }
}

```

```
[edit]
lab@r7# show protocols
isis {
  export ospf-isis;
  reference-bandwidth 500m;
  lsp-lifetime 3600;
  level 1 preference 155;
  interface all {
    hello-authentication-key "$9$j5kmTOBEhrv"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
    level 1 priority 0;
  }
}
ospf {
  external-preference 159;
  export isis-ospf;
  area 0.0.0.2 {
    nssa;
    authentication-type simple; # SECRET-DATA
    interface fe-0/3/3.0 {
      authentication-key "$9$6vcM/pBIRSeMXhS"; # SECRET-DATA
    }
  }
}
```

Spot the Issues: Review Questions

The following review questions are based on the topology shown earlier in Figure 4.4.

1. Will the following IS-IS configuration correctly configure interface fe-0/0/0 for IS-IS level 1 operation only?

```
[edit protocols isis]
lab@r5# show
interface fe-0/0/0.0 {
    level 1 hello-interval 10;
}
interface all {
    level 2 disable;
}
```

2. The following configuration results in an IS-IS adjacency to r3 and r4, but no routes are installed and the LSDB contains no LSP from these routers. What is a likely cause for this symptom?

```
[edit protocols isis]
lab@r5# show
export summ;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 2 {
    authentication-key "$9$qP5FREy8X-"; # SECRET-DATA
    authentication-type simple; # SECRET-DATA
}
level 1 preference 155;
interface fe-0/0/0.0 {
    level 2 disable;
    level 1 {
        hello-authentication-key "$9$H.fz1IcSeW"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
interface fe-0/0/1.0 {
    level 2 disable;
    level 1 {
        hello-authentication-key "$9$EORS1M2gJZjq"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
    }
}
```

```
interface all {
  level 1 disable;
  level 2 {
    hello-authentication-key "$9$5znCyrVMX-"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
  }
}
```

3. You have incorrectly specified the IP address for r5's at-0/12/1.35 interface, as highlighted next. Will an IS-IS adjacency still be possible?

```
[edit interfaces at-0/2/1]
lab@r5# show
atm-options {
  vpi 0 maximum-vcs 17;
  vpi 5 maximum-vcs 36;
  ilmi;
}
unit 35 {
  vci 5.35;
  shaping {
    cbr 50m;
  }
  oam-period 10;
  oam-liveness {
    up-count 2;
    down-count 2;
  }
  family inet {
    address 10.0.20.1/30;
  }
  family iso;
}
```

4. What is missing from the IS-IS configuration shown next?

```
[edit protocols isis]
lab@r5# show
export summ;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 2 {
  authentication-key "$9$jEkMTOBEhrv"; # SECRET-DATA
```

```

    authentication-type simple; # SECRET-DATA
  }
  level 1 preference 155;
  interface fe-0/0/0.0 {
    level 2 disable;
    level 1 {
      hello-authentication-key "$9$H.fz1IcSeW"; # SECRET-DATA
      hello-authentication-type md5; # SECRET-DATA
    }
  }
  interface fe-0/0/1.0 {
    level 2 disable;
    level 1 {
      hello-authentication-key "$9$EORS1M2gJZjq"; # SECRET-DATA
      hello-authentication-type md5; # SECRET-DATA
    }
  }
  interface at-0/2/1.35 {
    level 1 disable;
    level 2 {
      hello-authentication-key "$9$WcqXNbiHmPQn"; # SECRET-DATA
      hello-authentication-type md5; # SECRET-DATA
    }
  }
  interface as1.0 {
    level 2 {
      hello-authentication-key "$9$d7w2a5T3nCu"; # SECRET-DATA
      hello-authentication-type md5; # SECRET-DATA
    }
    level 1 disable;
  }
}

```

5. Your goal is to advertise a 10.0.4/22 summary to level 2 areas only. Will the following policy achieve this requirement?

```

[edit]
lab@r3# show policy-options
policy-statement summ {
  term 1 {
    from {
      route-filter 10.0.4.0/22 longer;
    }
  }
}

```

```

        to level 2;
        then reject;
    }
    term 2 {
        from {
            protocol aggregate;
            route-filter 10.0.4.0/22 exact;
        }
        then accept;
    }
}

```

6. You have decided to save some time by using a configuration group intended to configure the iso family on your router's interface with less typing. What could go wrong, based on the following configuration?

```

[edit]
lab@r2# show groups
iso {
    interfaces {
        <*> {
            unit <*> {
                family iso;
            }
        }
    }
}

```

```

[edit]
lab@r2# show interfaces apply-groups
apply-groups iso;

```

```

[edit]
lab@r2# show protocols isis
export direct;
reference-bandwidth 500m;
lsp-lifetime 3600;
level 1 wide-metrics-only;
interface all {
    hello-authentication-key "$9$TF6Ar1MWxd"; # SECRET-DATA
    hello-authentication-type md5; # SECRET-DATA
    level 2 disable;
}

```

Spot the Issues: Answers

1. No, this configuration will result in level 1 and level 2 operation on the fe-0/0/0 interface. The specific mention of interface fe-0/0/0 exempts it from the `interface all level 2 disable` statement. You will need to specifically disable level 2 operation on this interface.
2. The most likely cause is mismatched LSP authentication as configured under the level 2 stanza. Hello authentication must be properly configured for the adjacency to form. To test if this is the cause, you could delete backbone area authentication-related entries, perform protocol tracing, or configure the `no-authentication-check` option.
3. Depending on the JUNOS software version, the answer will be *yes* or *no*. The 5.2.x JUNOS software used in this test bed will permit the IS-IS adjacency to form. Unlike OSPF, the IS-IS protocol does not understand IP addressing, and can therefore allow an IS-IS adjacency when IP subnet parameters have been mismatched. This aspect of IS-IS is well worth noting, because troubleshooting issues that relate to simple IP addressing mistakes are made more difficult when the operator believes that the presence of an IS-IS adjacency provides vindication of the interface's IP properties.
4. The lack of an `interface all` term, and specific mention of the router's lo0 interface, means that the router will have no ISO NET (assuming you have correctly assigned the NET to the router's lo0 interface). The IS-IS protocol will not function without a NET. In this case, protocol tracing will report errors as shown next:

```
May 29 04:42:41 task_receive_packet: task IS-IS I/O from index 5 addr
0.a0.c9.69.c5.27 socket 11 length 62
May 29 04:42:41 ERROR: received IIH but have no local sysid .
```

5. No. The lack of a `to level 2` clause in term 2 will result in the summary route being advertised into all IS-IS levels.
6. Your configuration group's wild-card syntax will match on all router interfaces and logical units, which includes the router's fxp0 interface. The use of `interface all` in the IS-IS stanza will in turn enable IS-IS operation on all such ISO-capable interfaces. This can lead to the formation of IS-IS adjacencies over the OoB network, which has the potential of catastrophic disruption considering the black holes that will occur when a router forwards packets destined for a remote router over an fxp0 adjacency. To alleviate this problem, you could rewrite the wild-card expression to be more selective or you could specifically disable the router's fxp0 interface under the IS-IS stanza. Both approaches are demonstrated next in the context of r2's case study configuration. The first example has the wild-card expression carefully written such that it will match only on the correct router interfaces (fe-0/0/1 through fe-0/0/3):

```
[edit groups iso]
lab@r2# show
iso {
    interfaces {
        "<fe-[1-3]>" {
```

```

        unit <*> {
            family iso;
        }
    }
}

```

The second approach compensates for the use of a sloppy wild-card expression (<*>) by explicitly disabling interfaces that should not run IS-IS:

```

[edit]
lab@r2# show protocols
isis {
    export direct;
    reference-bandwidth 500m;
    lsp-lifetime 3600;
    level 1 wide-metrics-only;
    interface fe-0/0/0.0 {
        disable;
    }
    interface all {
        hello-authentication-key "$9$TF6Ar1MWxd"; # SECRET-DATA
        hello-authentication-type md5; # SECRET-DATA
        level 2 disable;
    }
    interface fxp0.0 {
        disable;
    }
}

```

