Chapter 5

Load Balancing in an MPLS Network

In an Multiprotocol Label Switched (MPLS) network, load balancing is the process of distributing traffic equally across lable switched paths. When you have added several LSPs to the same egress router, the default behavior of the JUNOS software is to select the LSP with the lowest metric to carry all traffic. If all of the LSPs have the same metric, one of the LSPs is selected at random and all traffic is forwarded over it. You can change this default behavior by configuring load balancing on an ingress router, allowing the JUNOS software to distribute the traffic equally across LSPs. (See Table 8.)

The terms *node* and *router* are used interchangeably throughout this book.

Table 8: Checklist for Load Balancing in an MPLS Network

Load Balancing in an MPLS Network Tasks	Command or Action
Load Balancing Overview on page 68	
Load-Balancing Options on page 69	The following load-balancing options are included: ■ IPv4 address family (INET) in the hash key ■ MPLS labels and IP payload in the hash key ■ LSP bandwidth
Configuring and Verifying Load Balancing on page 70	
1. Define a Load-Balancing Policy on page 70	[edit] edit policy-options [edit policy-options] set policy-statement *policy-name* then load-balance per-packet show commit
2. Apply the Load-Balancing Policy to the Forwarding Table on page 71	[edit] edit routing-options [edit routing-options] set forwarding-table export *policy-name* show commit
3. Verify That Load Balancing Is Working on page 72	show configuration show route show route forwarding-table show mpls lsp statistics monitor interface traffic clear mpls lsp statistics clear interface statistics

Load Balancing in an MPLS Network Tasks	Command or Action
Example: Load-Balanced MPLS Network on page 75	
Router Configurations for the Load-Balanced MPLS Network on page 76	show configuration | no-more
Using Hash-Key Load Balancing for LSP Traffic on page 87	
1. Configuring MPLS Labels and IP Payload to Load-Balance LSP Traffic on page 87	[edit] edit forwarding-options hash-key [edit forwarding-options hash-key] set family mpls label-1 set family mpls label-2 set family mpls payload ip show commit
2. Configuring the IPv4 Address Family to Load-Balance LSP Traffic on page 89	[edit] edit forwarding-options hash-key [edit forwarding-options hash-key] set family inet layer-3 set family inet layer-4 show commit
Hash Key Network Examples on page 91	
1. Example: Load-Balancing a Network with Aggregated Interfaces on page 92	
a. Verifying the Operation of Load Balancing with Aggregated Interfaces on page 93	show configuration forwarding-options show interfaces statistics *interface-name* detail show mpls lsp statistics
b. Router Configurations for the Aggregated Interfaces Network on page 96	show configuration | no-more
2. Example: Load-Balancing a Network Using INET in the Hash Key on page 103	
a. Verifying the Operation of INET Load Balancing on page 104	show configuration show route forwarding-table destination *destination* show route monitor interface traffic show mpls lsp statistics
b. Router Configurations for the INET Load-Balanced Network on page 107	show configuration | no-more
Using Bandwidth to Unevenly Load-Balance RSVP LSPs on page 117	
1. Configure Bandwidth to Unevenly Load-Balance Traffic on page 119	[edit] edit protocols mpls [edit protocols mpls] set label-switched-path *lsp-path-name* bandwidth *bps* show [edit protocols rsvp] set load-balance bandwidth show commit

Load Balancing in an MPLS Network Tasks	Command or Action
2. Verify the Operation of Uneven Bandwidth Load Balancing on page 120	show route protocol rsvp detail show mpls lsp statistics
3. Router Configurations for Bandwidth Load Balancing on page 122	show configuration | no-more
Traffic Flows Before Load Balancing on page 124	show route | find mpls monitor interface traffic show mpls lsp statistics

Load Balancing Overview

In an MPLS network, load balancing is generally configured on an ingress router. The load-balancing configuration distributes traffic equally across LSPs with a hash algorithm that selects the next-hop destination and installs it into the forwarding table for the active route of the LSP. Whenever the next hop changes in any way, the hash algorithm changes the next-hop address.

Use load balancing when you have many LSPs with equal-cost next hops going out different interfaces to the same destination. For example, ingress router R1 has four LSPs configured to egress router R3, transiting R2. All four LSPs have the same metric from R2 to R3, but exit R2 from different interfaces. Without load balancing configured, one of the LSPs is selected at random and all traffic is forwarded over it. With load balancing configured, traffic is balanced evenly across all four LSPs.

Depending on the version of the Internet Processor ASIC in the routing platform, a different method is used to load-balance traffic. Routing platforms with an Internet Processor I ASIC use a round-robin method, sending each packet over a different link. The round-robin method results in a good traffic balance at the cost of potential out-of-order packet arrival, which is not good for TCP.

Routing platforms with the Internet Processor II ASIC divide traffic into individual traffic flows across up to 16 next hops, keeping each individual flow on a single interface. A flow is comprised of packets with the following identical parameters:

- Source IP address
- Destination IP address
- Protocol
- Source port number
- Destination port number
- Source interface index
- Type of service (ToS)

For example, if there are 60,000 prefixes in a routing table, and 6 links between two routers, 10,000 prefixes will go across each link. In the core of the network, the law of averages produces good traffic load balancing. However, at the edge of the network, where there may not be a large number of prefixes, traffic may not be well balanced.

To summarize, an Internet Processor I ASIC spreads packets with the same parameters across multiple equal-cost next hops; while an Internet Processor II ASIC sends packets with the same parameters to the same next hop, since they are in the same flow. The JUNOS software command to turn on load balancing uses the action load-balance per-packet, which is misnamed in relation to the Internet Processor II ASIC. On the Internet Processor II ASIC, this command actually enables per-flow load balancing.

To configure load balancing, include a policy statement at the [edit policy-options] hierarchy level. This policy statement must be applied as an export policy at the [edit forwarding-options] hierarchy level. For more information, see "Configuring and Verifying Load Balancing" on page 70.

Load-Balancing Options

After you configure load balancing, if the outbound traffic across equal-cost next hops is not balanced to your satisfaction, you can provide additional information to identify traffic flows and balance traffic more evenly or unevenly, depending on your requirements. You can provide additional information to alter the way load balancing works in the following ways:

- Include MPLS labels and IP payload statements in the hash-key configuration at the [edit forwarding-options hash-key] hierarchy level to evenly balance traffic across LSPs and aggregated interfaces. For more information, see "Configuring MPLS Labels and IP Payload to Load-Balance LSP Traffic" on page 87.

- Include the IPv4 address family (INET) in the hash-key configuration at the [edit forwarding-options hash-key] hierarchy level to ensure that traffic is evenly load-balanced across LSPs and interfaces, and that packets in the same flow are sent out through the same interface. For more information, see "Configuring the IPv4 Address Family to Load-Balance LSP Traffic" on page 89.

- Include bandwidth at the [edit protocols mpls label-switched-path *lsp-name*] and the [edit protocols rsvp] hierarchy levels to change the number of prefixes carried by an LSP and thereby create an uneven load balance for different LSPs. For more information, see "Using Bandwidth to Unevenly Load-Balance RSVP LSPs" on page 117.

Configuring and Verifying Load Balancing

Purpose Load balancing is configured on the ingress router and uses the hash algorithm to distribute traffic equally across paths. The hash algorithm is designed to distribute packets to prevent any single link from being saturated. Before you can configure and verify load balancing in an MPLS network, you must have all the necessary MPLS components and protocols configured correctly. For information on configuring an MPLS network, see the *JUNOS MPLS Network Operations Guide*.

Keep the following information in mind when you configure load balancing:

- The load-balance per packet policy is configured on an ingress router with more than one LSP configured to the same egress router.
- Load balancing offers no guarantee of equal distribution of traffic over equal-cost links, nor does it guarantee that increasing the number of Internet flows will create a better hash distribution.

Steps To Take To configure and verify load balancing, follow these steps:

1. Define a Load-Balancing Policy on page 70
2. Apply the Load-Balancing Policy to the Forwarding Table on page 71
3. Verify That Load Balancing Is Working on page 72

Step 1: Define a Load-Balancing Policy

Purpose On the ingress or transit router, you can include a policy statement that performs load balancing on all routes. For information on including a policy statement that performs load balancing on specific routes, see "Configuring Per-Packet Load Balancing" in the *JUNOS Routing Protocols Configuration Guide*.

Action On the ingress or transit router, to define a load-balancing policy for all routes, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit policy-options
```

2. Define the load-balance policy and action:

```
[edit policy-options]
user@host# set policy-statement policy-name then load-balance per-packet
```

3. Verify and commit the configuration:

```
user@host# show
user@host# commit
```

Sample Output

```
user@R6> edit
Entering configuration mode

[edit]
user@R6# edit policy-options

[edit policy-options]
user@R6# set policy-statement load-balance-traffic then load-balance per-packet

[edit policy-options]
user@R6# show
policy-statement load-balance-traffic {
    then {
        load-balance per-packet;
    }
}

[edit policy-options]
user@R6# commit
commit complete
```

What It Means The sample output from ingress router R6 shows the process for configuring load balancing. On an Internet Processor I ASIC, packets with the same parameters are spread across multiple equal-cost next hops; while an Internet Processor II ASIC sends packets with the same parameters to the same next hop, since they are in the same flow. The JUNOS software command to turn on load balancing uses the action load-balance per-packet, which is misnamed in relation to the Internet Processor II ASIC. On the Internet Processor II ASIC, this command actually enables per-flow load balancing.

Step 2: Apply the Load-Balancing Policy to the Forwarding Table

Purpose Apply the policy configured in Step 1 to routes exported from the routing table to the forwarding table.

Action To apply a load-balancing policy to the forwarding table, follow these steps:

1. In configuration mode, go to the following hierarchy level:

   [edit]
   user@host# **edit routing-options**

2. Define a load-balance per packet action:

   [edit routing-options]
   user@host# **set forwarding-table export** *policy-name*

3. Verify and commit the configuration:

   user@host# **show**
   user@host# **commit**

Sample output

```
[edit]
user@R6# edit routing-options

[edit routing-options]
user@R6# set forwarding-table export load-balance-traffic

[edit routing-options]
user@R6# show
static {
[...Output truncated...]
}
router-id 192.168.6.1;
autonomous-system 65432;
forwarding-table {
    export load-balance-traffic;
}

[edit routing-options]
user@R6# commit
commit complete
```

What It Means The sample output shows the process for applying a load-balancing policy to export routes from the routing table to the forwarding table.

Step 3: Verify That Load Balancing Is Working

Purpose After configuring load balancing, check that traffic is load-balanced equally across paths. In this section, the command output reflects the load-balancing configuration of the example network shown in Figure 10 on page 75. The clear commands are used to reset LSP and interface counters to zero so that the values reflect the operation of the load-balancing configuration.

Action To verify load balancing across interfaces and LSPs, use the following command on the ingress router:

user@host# **show configuration**

To verify load balancing across interfaces and LSPs, use the following commands on a transit router:

user@host# **show route**
user@host# **show route forwarding-table**
user@host# **show mpls lsp statistics**
user@host# **monitor interface traffic**
user@host# **clear mpls lsp statistics**
user@host# **clear interface statistics**

Sample Output The following sample output is for the configuration on ingress router R1:

```
user@R1> show configuration | no-more
[...Output truncated...]
routing-options {
    [...Output truncated...]
    forwarding-table {
        export lbpp;
    }
}
```

```
[...Output truncated...]
policy-options {
    policy-statement lbpp {
        then {
            load-balance per-packet;
        }
    }
}
```

What It Means The sample output for the show configuration command on ingress router R1 shows that load balancing is correctly configured with the lbpp policy statement. Also, the lbpp policy is exported into the forwarding table at the [edit routing-options] hierarchy level.

Sample Output The following sample output is from transit router R2:

```
user@R2> show route 192.168.0.1 terse

inet.0: 25 destinations, 27 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 192.168.0.1/32     O  10          3             so-0/0/1.0
                                                 >so-0/0/2.0
[...Output truncated...]
```

What It Means The sample output for the show route command issued on transit router R2 shows the two equal-cost paths (so-0/0/1 and so-0/0/2) through the network to the loopback address to R0 (192.168.0.1). Even though the right angle bracket (>) usually indicates the active route, in this instance it does not, as shown in the following four sample outputs.

Sample Output The following sample output is from transit router R2:

```
user@R2> monitor interface traffic

R2                                  Seconds: 65                   Time: 11:41:14

Interface    Link  Input packets        (pps)     Output packets        (pps)
 so-0/0/0     Up               0          (0)                  0          (0)
 so-0/0/1     Up             126          (0)             164659       (2128)
 so-0/0/2     Up           85219       (1004)             164598       (2128)
 so-0/0/3     Up               0          (0)                  0          (0)
 fe-0/1/0     Up          328954       (4265)              85475       (1094)
 fe-0/1/1     Up               0          (0)                  0          (0)
 fe-0/1/2     Up               0          (0)                  0          (0)
 fe-0/1/3     Up               0          (0)                  0          (0)
[...Output truncated...]
```

What It Means The sample output for the monitor interface traffic command issued on transit router R2 shows that output traffic is evenly distributed across the two interfaces so-0/0/1 and so-0/0/2.

Sample Output The following sample output is from transit router R2:

```
user@R2> show mpls lsp statistics
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 5 sessions
To              From            State     Packets            Bytes LSPname
192.168.0.1     192.168.1.1     Up          87997         17951388 lsp1
192.168.0.1     192.168.1.1     Up          87997         17951388 lsp2
192.168.0.1     192.168.1.1     Up          87997         17951388 lsp3
192.168.0.1     192.168.1.1     Up          87997         17951388 lsp4
192.168.6.1     192.168.0.1     Up              0                0 r0-r1
Total 5 displayed, Up 5, Down 0
```

What It Means The sample output for the show mpls lsp statistics command issued on transit router R2 shows that output traffic is evenly distributed across the four LSPs configured on ingress router R6.

Sample Output The following sample output is from transit router R2:

```
user@R2> show route forwarding-table destination 10.0.90.14
Routing table: inet
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
10.0.90.12/30      user     0                    ulst 262144     6
                                                 ucst   345     5 so-0/0/1.0
                                                 ucst   339     2 so-0/0/2.0
```

What It Means The sample output for the show route forwarding-table destination command issued on transit router R2 shows ulst in the Type field, which indicates that load balancing is working. The two unicast (ucst) entries in the Type field are the two next hops for the LSPs.

Sample Output The following sample output is from transit router R2:

```
user@R2> show route forwarding-table | find mpls
Routing table: mpls
MPLS:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            perm     0                    dscd    38     1
0                  user     0                    recv    37     3
1                  user     0                    recv    37     3
2                  user     0                    recv    37     3
100112             user     0                   Swap 100032       so-0/0/1.0
100128             user     0                   Swap 100048       so-0/0/1.0
100144             user     0 10.0.12.13        Swap 100096       fe-0/1/0.0
100160             user     0                   Swap 100112       so-0/0/2.0
100176             user     0                   Swap 100128       so-0/0/2.0
```

What It Means The sample output for the show route forwarding-table | find mpls command issued on transit router R2 shows the MPLS routing table that contains the labels received and used by this router to forward packets to the next-hop router. This routing table is used mostly on transit routers to route packets to the next router along an LSP. The first three labels in the Destination column (Label 0, Label 1, and Label 2) are automatically entered by MPLS when the protocol is enabled. These labels are reserved MPLS labels defined in RFC 3032. Label 0 is the IPv4 explicit null label. Label 1 is the MPLS equivalent of the IP Router Alert label, and Label 2 is the IPv6 explicit null label.

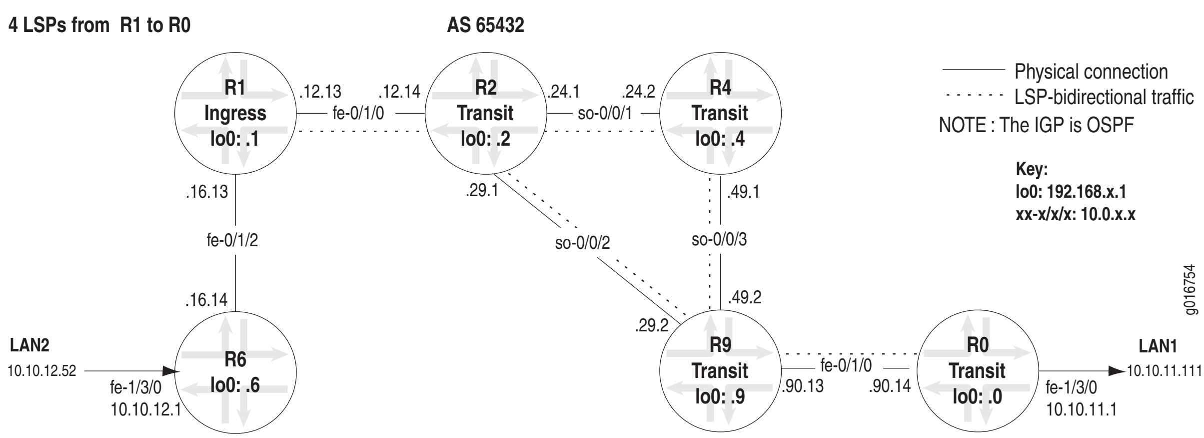
The remaining five labels in the Destination column are nonreserved labels that the router uses to forward traffic, and the last column Netif, shows the interfaces used to send the labeled traffic. For nonreserved labels, the second Type column shows the operation performed on matching packets. In this example, all non-reserved packets are swapped for outgoing packet labels. For example, packets with the label 100112 have their label swapped for 100032 before they are pushed out of interface so-0/0/1.0.

Example: Load-Balanced MPLS Network

Purpose When you configure several RSVP LSPs to the same egress router, the LSP with the lowest metric is selected and carries all traffic. If all of the LSPs have the same metric, one of the LSPs is selected at random and all traffic is forwarded over it. To distribute traffic equally across all LSPs, you can configure load balancing on the ingress or transit routers, depending on the type of load balancing configured.

Figure 10 illustrates an MPLS network with four LSPs configured to the same egress router (R0). Load balancing is configured on ingress router R1. The example network uses Open Shortest Path First (OSPF) as the interior gateway protocol (IGP) with OSPF area 0.0.0.0. An IGP is required for the Constrained Shortest Path First (CSPF) LSP, which is the default for the JUNOS software. In addition, the example network uses a policy to create BGP traffic.

Figure 10: Load-Balancing Network Topology



The network shown in Figure 10 on page 75 consists of the following components:

- A full-mesh interior BGP (IBGP) topology, using AS 65432
- MPLS and RSVP enabled on all routers
- A send-statics policy on routers R1 and R0 that allows a new route to be advertised into the network
- Four unidirectional LSPs between R1 and R0, and one reverse direction LSP between R0 and R1, which allows for bidirectional traffic
- Load balancing configured on ingress router R1

The network shown in Figure 10 on page 75 is a BGP full-mesh network. Since route reflectors and confederations are not used to propagate BGP learned routes, each router must have a BGP session with every other router running BGP.

For complete configurations for all routers in the example MPLS network, see "Router Configurations for the Load-Balanced MPLS Network" on page 76.

For a description of the situation before and after load balancing is configured in the network to use all four LSPs to forward traffic, see "Traffic Flows Before Load Balancing" on page 124.

Router Configurations for the Load-Balanced MPLS Network

Purpose The configurations in this section are for the six routers in the example network illustrated in Figure 10 on page 75.

Action To display the configuration of a router, use the following JUNOS CLI operational mode command:

user@host> **show configuration | no-more**

Sample Output 1 The following configuration output is for edge router R6.

```
user@R6> show configuration | no-more
[...Output truncated...]
interfaces {
    fe-0/1/2 {
        unit 0 {
            family inet {
                address 10.0.16.14/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    fe-1/3/0 {
        unit 0 {
            family inet {
                address 10.10.12.1/24;
            }
        }
    }
    fxp0 {
        unit 0 {
```

```
                family inet {
                    address 192.168.70.148/21;
                }
            }
        }
        lo0 {
            unit 0 {
                family inet {
                    address 192.168.6.1/32;
                }
            }
        }
    }
    routing-options {
        static {
    [...Output truncated...]
        router-id 192.168.6.1; #Manually configured RID
        autonomous-system 65432; #Full mesh IBGP
        }
    }
    protocols {
        rsvp {
            interface fe-0/1/2.0;
            interface fxp0.0 {
                disable;
            }
        }
        mpls {
            interface fe-0/1/2.0;
            interface fxp0.0 {
                disable;
            }
        }
        bgp {
            group internal {
                type internal;
                local-address 192.168.6.1;
                neighbor 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.9.1;
                neighbor 192.168.0.1;
            }
        }
        ospf { #IGP enabled
            traffic-engineering;
            area 0.0.0.0 {
                interface fe-0/1/2.0;
                interface fe-1/3/0.0;
                interface lo0.0 {
                    passive; #Ensures protocols do not run over this interface
                }
            }
        }
    }
```

Sample Output 2 The following configuration output is for ingress router R1.

```
user@R1> show configuration | no-more
[...Output truncated...]
interfaces {
    fe-0/1/0 {
```

```
            unit 0 {
                family inet {
                    address 10.0.12.13/30;
                }
                family mpls; #MPLS enabled on relevant interfaces
            }
        }
        fe-0/1/2 {
            unit 0 {
                family inet {
                    address 10.0.16.13/30;
                }
                family mpls;
            }
        }
        fxp0 {
            unit 0 {
                family inet {
                    address 192.168.70.143/21;
                }
            }
        }
        lo0 {
            unit 0 {
                family inet {
                    address 192.168.1.1/32;
                }
            }
        }
    }
    routing-options {
        static {
            [...Output truncated...]
        route 100.100.1.0/24 reject; #Static route for send-statics policy
    }
        router-id 192.168.1.1; #Manually configured RID
        autonomous-system 65432; #Full mesh IBGP
        forwarding-table {
            export lbpp; #Routes exported to forwarding table
        }
    }
    protocols {
        rsvp {
            interface fe-0/1/0.0;
            interface fe-0/1/2.0;
            interface fxp0.0 {
                disable;
            }
        }
        mpls {
            label-switched-path lsp1 { #First LSP
                to 192.168.0.1; # Destination of the LSP
                install 10.0.90.14/32 active; # The prefix is installed in the
                primary via-r4;               # inet.0 routing table
            }
            label-switched-path lsp2 {
                to 192.168.0.1;
                install 10.0.90.14/32 active;
                primary via-r2;
            }
            label-switched-path lsp3 {
                to 192.168.0.1;
                install 10.0.90.14/32 active;
```

```
                primary via-r2;
            }
            label-switched-path lsp4 {
                to 192.168.0.1;
                install 10.0.90.14/32 active;
                primary via-r4;
            }
            path via-r2 { #Primary path to spread traffic across interfaces
                10.0.29.2 loose;
            }
            path via-r4 {
                10.0.24.2 loose;
            }
            interface fe-0/1/0.0;
            interface fe-0/1/2.0;
            interface fxp0.0 {
                disable;
            }
        }
        bgp {
            export send-statics; #Allows advertising of a new route
            group internal {
                type internal;
                local-address 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.9.1;
                neighbor 192.168.6.1;
                neighbor 192.168.0.1;
            }
        }
        ospf { #IGP enabled
            traffic-engineering;
            area 0.0.0.0 {
                interface fe-0/1/0.0;
                interface fe-0/1/2.0;
                interface lo0.0 {
                    passive; #Ensures protocols do not run over this interface
                }
            }
        }
    }
    policy-options { #Load balancing policy
        policy-statement lbpp {
            then {
                load-balance per-packet;
            }
        }
        policy-statement send-statics { #Static route policy
            term statics {
                from {
                    route-filter 100.100.1.0/24 exact;
                }
                then accept;
            }
        }
    }
```

Sample Output 3 The following configuration output is for transit router R2.

```
user@R2> show configuration | no-more
[...Output truncated...]
```

```
interfaces {
    so-0/0/1 {
        unit 0 {
            family inet {
                address 10.0.24.1/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.0.29.1/30;
            }
            family mpls;
        }
    }
    fe-0/1/0 {
        unit 0 {
            family inet {
                address 10.0.12.14/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.144/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.2.1/32;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
    router-id 192.168.2.1; #Manually configured RID
    autonomous-system 65432; #Full mesh IBGP
    }
}
protocols {
    rsvp {
        interface so-0/0/1.0;
        interface fe-0/1/0.0;
        interface so-0/0/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface fe-0/1/0.0;
        interface so-0/0/1.0;
        interface so-0/0/2.0;
        interface fxp0.0 {
            disable;
        }
```

```
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.2.1;
            neighbor 192.168.1.1;
            neighbor 192.168.4.1;
            neighbor 192.168.9.1;
            neighbor 192.168.6.1;
            neighbor 192.168.0.1;
        }
    }
    ospf { #IGP enabled
        traffic-engineering;
        area 0.0.0.0 {
            interface fe-0/1/0.0;
            interface so-0/0/1.0;
            interface so-0/0/2.0;
            interface lo0.0 {
                passive; #Ensures protocols do not run over this interface
            }
        }
    }
}
```

Sample Output 4 The following configuration output is for transit router R4.

```
user@R4> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/1 {
        unit 0 {
            family inet {
                address 10.0.24.2/30;
            }
            family mpls; # MPLS enabled on relevant interfaces
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.0.49.1/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.146/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.4.1/32;
            }
        }
    }
}
routing-options {
```

```
    static {
        [...Output truncated...]
    router-id 192.168.4.1; #Manually configured RID
    autonomous-system 65432; #Full mesh IBGP
}
protocols {
    rsvp {
        interface so-0/0/1.0;
        interface so-0/0/3.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface so-0/0/1.0;
        interface so-0/0/3.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.4.1;
            neighbor 192.168.1.1;
            neighbor 192.168.2.1;
            neighbor 192.168.9.1;
            neighbor 192.168.6.1;
            neighbor 192.168.0.1;
        }
    }
    ospf { #IGP enabled
        traffic-engineering;
        area 0.0.0.0 {
            interface so-0/0/1.0;
            interface so-0/0/3.0;
            interface lo0.0 {
                passive; #Ensures protocols do not run over this interface
            }
        }
    }
}
```

Sample Output 5 The following configuration output is for transit router R9.

```
user@R9> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.0.29.2/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.0.49.2/30;
            }
            family mpls;
```

```
            }
        }
        fe-0/1/0 {
            unit 0 {
                family inet {
                    address 10.0.90.13/30;
                }
                family mpls;
            }
        }
        fxp0 {
            unit 0 {
                family inet {
                    address 192.168.69.206/21;
                }
            }
        }
        lo0 {
            unit 0 {
                family inet {
                    address 192.168.9.1/32;
                }
            }
        }
    }
    routing-options {
        static {
            [...Output truncated...]
        router-id 192.168.9.1; #Manually configured RID
        autonomous-system 65432; #Full mesh IBGP
    }
    protocols {
        rsvp {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface fxp0.0 {
                disable;
            }
        }
        mpls {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface fxp0.0 {
                disable;
            }
        }
        bgp {
            group internal {
                type internal;
                local-address 192.168.9.1;
                neighbor 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.0.1;
                neighbor 192.168.6.1;
            }
        }
        ospf { #IGP enabled
            traffic-engineering;
            area 0.0.0.0 {
                interface so-0/0/2.0;
```

```
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface lo0.0 {
                passive; #Ensures protocols do not run over this interface
            }
        }
    }
}
```

Sample Output 6 The following configuration output is for egress router R0.

```
user@R0> show configuration | no-more
[...Output truncated...]
interfaces {
    fe-0/1/0 {
        unit 0 {
            family inet {
                address 10.0.90.14/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    fe-1/3/0 {
        unit 0 {
            family inet {
                address 10.10.11.1/24;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.69.207/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.0.1/32;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
        route 100.100.10.0/24 reject; #Static route for send-statics policy
    }
    router-id 192.168.0.1; #Manually configured RID
    autonomous-system 65432; #Full mesh IBGP
}
protocols {
    rsvp {
        interface fe-0/1/0.0;
        interface fe-1/3/0.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path r0-r6 {
            to 192.168.6.1;
```

```
        }
        interface fe-0/1/0.0;
        interface fe-1/3/0.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.0.1;
            export send-statics; #Allows advertising of a new route
            neighbor 192.168.9.1;
            neighbor 192.168.6.1;
            neighbor 192.168.1.1;
            neighbor 192.168.2.1;
            neighbor 192.168.4.1;
        }
    }
    ospf { #IGP enabled
        traffic-engineering;
        area 0.0.0.0 {
            interface fe-0/1/0.0;
            interface fe-1/3/0.0;
            interface lo0.0 {
                passive; #Ensures protocols do not run over this interface
            }
        }
    }
}
policy-options {
    policy-statement send-statics {
        term statics {
            from {
                route-filter 100.100.10.0/24 exact;
            }
            then accept;
        }
    }
}
```

What It Means Sample Outputs 1 through 6 show the base interfaces, routing options, protocols, and policy options configurations for all six routers in the example network illustrated in Figure 10 on page 75.

All routers in the network have MPLS, RSVP, and BGP enabled. OSPF is configured as the IGP, and relevant interfaces have basic IP information and MPLS support.

In addition, all routers have the router ID (RID) configured manually at the [edit routing-options] hierarchy level to avoid duplicate RID problems. The passive statement is included in the OSPF configuration to ensure that protocols are not run over the loopback (lo0) interface and that the loopback (lo0) interface is advertised correctly throughout the network.

Sample Outputs 1, 3, 4, and 5 for R6, R2, R4, and R9 show the base configuration for transit label-switched routers. The base configuration includes all interfaces enabled for MPLS, the RID manually configured, and the relevant protocols (RSVP, MPLS, BGP, and OSPF).

Sample Output 2 from ingress router R1 shows the base configuration plus four LSPs (lsp1 through lsp4) configured to R0. The four LSPs are configured with different primary paths that specify a loose hop through R4 for lsp1 and lsp4, and through R2 for lsp2 and lsp3.

To create traffic, R1 has a static route (100.100.1.0/24) configured at the [edit routing-options static route] hierarchy level. The prefix is included in the send-statics policy at the [edit policy-options send statics] hierarchy level so the routes can become BGP routes.

In addition, on the ingress router R1, load balancing is configured using the per-packet option, and the policy is exported at the [edit routing-options forwarding-table] hierarchy level.

Sample Output 6 from egress router R0 shows one LSP (r0-r6) to R6 used to create bidirectional traffic. OSPF requires bidirectional LSP reachability before it will advertise the LSP into the IGP. Although the LSP is advertised into the IGP, no hello messages or routing updates occur over the LSP—only user traffic is sent over the LSP. The router uses its local copy of the IGP database to verify bidirectional reachability.

In addition, R0 has a static route (100.100.10.0/24) configured at the [edit routing-options static route] hierarchy level. The prefix is included in the send-statics policy at the [edit policy-options send statics] hierarchy level so the routes can become BGP routes.

Using Hash-Key Load Balancing for LSP Traffic

Purpose

If the outbound traffic across equal-cost next hops is not well balanced after you have load balancing configured, you can configure the hash key to provide additional information to further identify traffic flows and balance traffic more evenly. The hash key is configured at the [edit forwarding-options] hierarchy level on ingress and transit routers, depending on your network configuration.

Within the hash key, you can configure the IPv4 address family (INET) or the MPLS protocol family. Typically for the best results, you configure the INET on the ingress router and the MPLS protocol family on the transit router. If a router happens to be both an ingress and transit router, you can configure both the INET and the MPLS protocol family. However, the INET will only be used when the router is acting as an ingress router and the MPLS protocol family will only be used when the router is acting as a transit router. You can configure only the INET on the ingress router, and check that the results are what is intended. Similarly, you can configure only MPLS labels on the transit router, and check the results.

In addition, the MPLS protocol family is most useful in configurations with aggregated interfaces, although it can be used on transit routers with regular (non-aggregated) interfaces.

Steps To Take

To use the hash key to load-balance LSP traffic, follow these steps:

1. Configuring MPLS Labels and IP Payload to Load-Balance LSP Traffic on page 87

2. Configuring the IPv4 Address Family to Load-Balance LSP Traffic on page 89

Step 1: Configuring MPLS Labels and IP Payload to Load-Balance LSP Traffic

If the outbound traffic across equal-cost next hops is not well balanced after you have load balancing configured, you can use MPLS labels and IP payload to provide additional information to further identify traffic flows and balance traffic more evenly, particularly between aggregated interfaces. With an aggregated interface, when you configure load balancing using the per-packet statement, the JUNOS software uses the first MPLS label in the hash algorithm to determine the next hop for the LSP. This behavior can result in an uneven distribution of traffic for aggregated interfaces.

You configure MPLS labels on a transit router because the transit router uses MPLS labels to forward traffic. Configuring the first two MPLS labels and the IP header is useful in the following circumstances:

- If there are many circuit cross-connect (CCC) MPLS LSPs using remote interface switching over an aggregated interface, configuring the first label can load-balance traffic between the component links of an aggregated interface. However, in other circumstances, such as an RSVP LSP, there is no benefit in configuring the first MPLS label by itself because load balancing using the per-packet statement uses the first label by default.

- If there are many CCC MPLS LSPs using remote interface switching over an aggregated interface with Martini Layer 1 VPN or Layer 2 VPN traffic, configuring the second MPLS label can load-balance traffic between component links of an aggregated interface.

- If there are CCC MPLS LSPs using remote interface switching over an aggregated interface with Layer 3 VPN traffic, Layer 2 VPN, or Martini Layer 2 VPN translational cross-connect (TCC) traffic, configuring the first and second MPLS labels and IP payload can balance traffic between component links of an aggregated interface.

Essentially, load balancing is similar across platforms even though there are slight differences between platforms. On M-series platforms, only Label 1 and the IP payload are used in the hash-key algorithm. On T-series platforms and the M320, all three labels (Label 1, Label 2, and IP payload) are used in the hash-key algorithm. However, there is no harm done if you configure all three labels on an M-series router. The router simply ignores Label 2.

Before you use MPLS labels and IP payload to load-balance traffic, you must have the load-balance per-packet statement configured at the [edit policy-options] hierarchy level and that policy applied as an export policy at the [edit forwarding-options] hierarchy level. For more information about configuring load balancing, see "Configuring and Verifying Load Balancing" on page 70.

Action To configure the hash key to load-balance LSP traffic, follow these steps:

1. Ensure that you have load balancing configured; see "Configuring and Verifying Load Balancing" on page 70.

2. In configuration mode, go to the following hierarchy level:

   [edit]
   user@host# **edit forwarding-options hash-key**

3. Depending on your network configuration, include a combination of MPLS labels to include in the configuration:

   [edit forwarding-options hash-key]
   user@host# **set family mpls label-1**
   user@host# **set family mpls label-2**
   user@host# **set family mpls payload ip**

NOTE: The configuration of all three statements together can be used on T-series and M320 routing platforms only. If you configure all three statements on an M-series router, only label-1 and the IP payload are used in the hash key.

---

4. Verify and commit the configuration:

   user@host# **show**
   user@host# **commit**

Sample Output

```
user@R2> edit
Entering configuration mode

[edit]
user@R2# edit forwarding-options hash-key

[edit forwarding-options hash-key]
user@R2# set family mpls label-1

[edit forwarding-options hash-key]
```

```
user@R2# set family mpls label-2

[edit forwarding-options hash-key]
user@R2# set family mpls payload ip

[edit forwarding-options hash-key]
user@R2# show
family mpls {
    label-1;
    label-2;
    payload {
        ip;
    }
}

[edit forwarding-options hash-key]
user@R2# commit
commit complete
```

What It Means The sample output shows the configuration of all three MPLS labels and verification that the configuration is correct.

Step 2: Configuring the IPv4 Address Family to Load-Balance LSP Traffic

Purpose If the outbound traffic across equal-cost next hops is not well balanced after you have load balancing configured, you can use the IPv4 address family (INET) to provide additional information to identify traffic flows and balance traffic more evenly. You configure the INET or port data on an ingress router. Configuring port data is useful if you are using TCP or UDP. However, it may not be useful to include port data when you are using protocols that are not associated with a Layer 4 port, for example, Layer 2 VPNs, generic routing encapsulation (GRE) tunneling, or Internet Control Message Protocol (ICMP).

To configure port data, you include the layer-3 or layer-4 options under the family-inet statement at the [edit forwarding-options hash-key] hierarchy level. When you include the layer-4 option, you must also include the layer-3 statement. If you omit the layer-3 statement, the management process removes the hash-key statement from the configuration and the router works as if you specified layer-3.

If you specify only the layer-3 statement in the configuration, the router uses the incoming interface index as well as the following Layer 3 information in the packet header to load-balance:

- Source IP address
- Destination IP address
- Protocol

If you include both the layer-3 and layer-4 statements, the router uses the following Layer 3 and Layer 4 information to load-balance:

- Source IP address
- Destination IP address
- Protocol

- Source port number
- Destination port number
- Incoming interface index

The router recognizes packets in which all of these Layer 3 and Layer 4 parameters are identical, and ensures that these packets are sent out through the same interface. This prevents problems that might otherwise occur with packets arriving at their destination out of their original sequence.

Before you use port data to send packets through the same interface, you must have the load-balance per-packet statement configured at the [edit policy-options] hierarchy level and that policy applied as an export policy at the [edit forwarding-options] hierarchy level. For more information about configuring load balancing, see "Configuring and Verifying Load Balancing" on page 70.

Action To configure the hash key with port data, follow these steps:

1. Ensure that you have load balancing configured, see "Configuring and Verifying Load Balancing" on page 70.
2. In configuration mode, go to the following hierarchy level:

   [edit]
   user@host# **edit forwarding-options hash-key**

3. Include Layer 3 (IP) data in the hash key:

   [edit forwarding-options hash-key]
   user@host# **set family inet layer-3**

4. Include Layer 4 TCP or UDP data in the hash key:

   [edit forwarding-options hash-key]
   user@host# **set family inet layer-4**

5. Verify and commit the configuration:

   user@host# **show**
   user@host# **commit**

Sample Output

```
user@R6> edit
Entering configuration mode

[edit]
user@R1# edit forwarding-options hash-key

[edit forwarding-options hash-key]
user@R1# set family inet layer-3

[edit forwarding-options hash-key]
user@R1# set family inet layer-4
```

```
[edit forwarding-options hash-key]
user@R1# show
family inet {
    layer-3;
    layer-4;
}

[edit forwarding-options hash-key]
user@R1# commit
commit complete
```

What It Means The sample output shows both the layer-3 and layer-4 options included in the hash key. Including both options provides additional information to identify traffic flows and balance traffic more evenly.

Hash Key Network Examples

Depending on the fields used in the hash-key algorithm and your network requirements, you can fine-tune the way traffic is load-balanced across your network. For example, if your network supports a large number of uses on routers running Network Address Translation (NAT) or Port Address Translation (PAT), the flows will be similar at Layer 3, so adding both Layer 3 and Layer 4 to the hash key can provide better load balancing. However, if a core router in your network is supporting tens of thousands of unrelated flows that vary significantly in source or destination addresses and incoming interfaces, including only Layer 3 in the hash key would probably result in a good distribution of traffic. With some exceptions, the more fields included in the hash-key algorithm, the greater the chance that traffic is unique, resulting in an optimal balance of traffic.

The following network examples illustrate various ways of using the hash key to load-balance traffic in different types of networks:

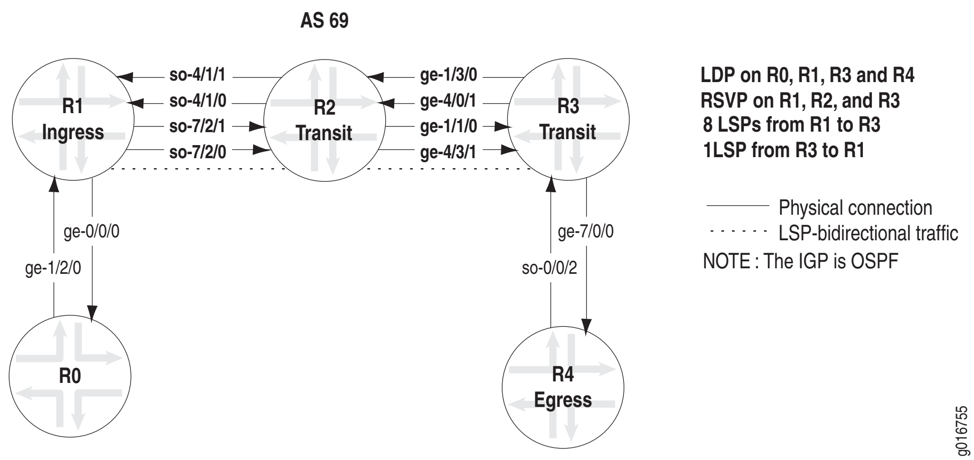
- Example: Load-Balancing a Network with Aggregated Interfaces on page 92
- Example: Load-Balancing a Network Using INET in the Hash Key on page 103

Example: Load-Balancing a Network with Aggregated Interfaces

Purpose With an aggregated interface, when you configure load balancing using the per-packet statement, the JUNOS software uses the first MPLS label in the hash algorithm to determine the next hop for the LSP. This behavior can result in an uneven distribution of traffic for aggregated interfaces.

This example describes load balancing using an LDP tunneled over RSVP on a network comprised of M-series and T-series routers with aggregated interfaces. Figure 11 illustrates the network used in this section.

Figure 11: Aggregated Interfaces Network Topology



The network topology in Figure 11 illustrates a router-only network with aggregated SONET and Ethernet interfaces that consists of the following components:

- BGP configured on PE routers R0 and R4
- LDP running on R0, R1, R3, and R4
- RSVP running on R1, R2, and R3
- LSPs set up from R1 to R3, and R3 to R1
- Aggregated interfaces on R1, R2, and R3
- The hash key configured on transit router R2
- Load balancing configured on R1

With the hash key configuration on R2, outbound traffic for the aggregated interface varies in terms of Label 1, Label 2, or IP payload. This variance in traffic should result in the equal distribution of traffic across different physical links of the aggregated interface.

The following information is included in this example:

- Verifying the Operation of Load Balancing with Aggregated Interfaces on page 93
- For the configuration output of all routers in this network, see "Router Configurations for the Aggregated Interfaces Network" on page 96

Verifying the Operation of Load Balancing with Aggregated Interfaces

Purpose On an M-series or T-series platform, when you configure only the first two labels and the labels vary between traffic flows, traffic may be distributed. However, if there is not much variation between the two labels, traffic may not be distributed equally across aggregated interfaces.

The following output illustrates two situations. The first example output shows a situation in which traffic is not balanced across interfaces because there is not enough variation between the two configured labels. The second example output shows a situation in which traffic is balanced; all three MPLS labels are configured and the third label has enough variation to yield good load-balancing results.

Action To verify the operation of the hash key, enter the following JUNOS CLI operational mode commands:

user@R2> **show configuration forwarding-options**
user@R2> **show interfaces statistics** *interface-name* **detail**
user@R2> **show mpls lsp statistics**

Sample Output

```
user@R2> show configuration forwarding-options
hash-key {
    family mpls {
        label-1;
        label-2;
    } # The IP payload option is missing.
}

user@R2> show interfaces statistics ae0 detail
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 289, Generation: 129
  Link-level type: Ethernet, MTU: 1514, Speed: 2000mbps, Loopback: Disabled,
Source filtering: Disabled,
  Flow control: Disabled, Minimum links needed: 1
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:90:69:0f:07:f0, Hardware address: 00:90:69:0f:07:f0
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :             102162                  560 bps
   Output bytes  :          166100728             30744472 bps
   Input  packets:               1259                    0 pps
   Output packets:            2442317                56515 pps
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                  0                    0 bps
   Input  packets:                  0                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed
discards: 0, Resource errors: 0
```

```
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0
  Ingress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                    0                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                   0                    0                    0
  Egress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort              2440822              2440822                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                1225                 1225                    0

  Logical interface ae0.0 (Index 66) (SNMP ifIndex 290) (Generation 131)
    Flags: SNMP-Traps Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :          1259          0        102162          560
        Output:       2441888      56515     166056858     30744472
    Link:
      ge-1/1/0.0
        Input :          1259          0        102162          560
        Output:       1488498      56515     101217864     30744272
      ge-4/3/1.0
        Input :             0          0             0            0
        Output:        953235          0      64822700          200
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/1/0.0                 0           0            0            0
      ge-4/3/1.0                 0           0            0            0
    Protocol inet, MTU: 1500, Generation: 128, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.35.200.4/30, Local: 10.35.200.5, Broadcast: 10.35.200.7,
Generation: 147
    Protocol iso, MTU: 1497, Generation: 128, Route table: 0
      Flags: Is-Primary
    Protocol mpls, MTU: 1488, Generation: 128, Route table: 0
      Flags: Is-Primary
```

What It Means

This sample shows that the hash key at the [edit forwarding-options] hierarchy level does not include the IP payload, indicating that the distribution of traffic is varying only by IP payload. The output for the show interfaces statistics command shows that traffic is not evenly distributed between links ge-1/1/0 and ge-4/3/1 of the aggregated interface ae0, probably due to the missing IP payload label.

Sample Output

```
user@R2> show configuration forwarding-options
hash-key {
    family mpls {
        label-1;
        label-2;
        payload {
            ip;
        }
    }
}

user@R2> show interfaces statistics ae0 detail
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 185, SNMP ifIndex: 289, Generation: 186
```

```
  Link-level type: Ethernet, MTU: 1514, Speed: 2000mbps, Loopback: Disabled,
Source filtering: Disabled,
  Flow control: Disabled, Minimum links needed: 1
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:90:69:0f:07:f0, Hardware address: 00:90:69:0f:07:f0
  Last flapped   : Never
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :              1000775                    0 bps
   Output bytes  :             79662734             30743104 bps
   Input  packets:                13273                    0 pps
   Output packets:              1168916                56512 pps
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed
discards: 0, Resource errors: 0
  Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors: 0
  Ingress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                    0                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                   0                    0                    0
  Egress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort              1186217              1186219                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont               13057                13057                    0

  Logical interface ae0.0 (Index 71) (SNMP ifIndex 292) (Generation 137)
    Flags: SNMP-Traps Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :         13273          0       1000775            0
        Output:       1168916      56512      79662734     30743104
    Link:
      ge-1/1/0.0 #Packets are evenly distributed across aggregated interfaces
        Input :         13273          0       1000775            0
        Output:        610927      28256      41716580     15371728
      ge-4/3/1.0
        Input :             0          0             0            0
        Output:        557989      28256      37946154     15371376
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/1/0.0                 0           0            0            0
      ge-4/3/1.0                 0           0            0            0
    Protocol inet, MTU: 1500, Generation: 146, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.35.200.4/30, Local: 10.35.200.5, Broadcast: 10.35.200.7,
Generation: 151
    Protocol iso, MTU: 1497, Generation: 147, Route table: 0
      Flags: Is-Primary
    Protocol mpls, MTU: 1488, Generation: 148, Route table: 0
      Flags: Is-Primary
```

What It Means This sample output shows the configuration of the hash key and the interface statistics for the aggregated interface ae0. The hash-key configuration specifies the labels used for outbound traffic on different physical links of the aggregated interface. In this case, two labels and IP payload are included in the configuration. The sample output for the show interface statistics command shows the outgoing traffic rate, which is evenly distributed between links ge-1/1/0 and ge-4/3/1 of aggregated interface ae0. However, an even distribution may not always be the case because it depends on a lot of factors, which can be defined at the [edit forwarding-options] hierarchy level.

Sample Output

```
user@R2> show mpls lsp statistics
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 9 sessions
To              From            State     Packets          Bytes LSPname
10.255.70.186   10.255.71.199   Up           1430         101943 to_R1_from_R3
10.255.71.199   10.255.70.186   Up          81745        5558550 to_R3_from_R1
10.255.71.199   10.255.70.186   Up          81748        5558768 to_R3_from_R1_1
10.255.71.199   10.255.70.186   Up          81760        5559492 to_R3_from_R1_2
10.255.71.199   10.255.70.186   Up         153259       10421488 to_R3_from_R1_3
10.255.71.199   10.255.70.186   Up         163509       11118573 to_R3_from_R1_4
10.255.71.199   10.255.70.186   Up         163453       11114666 to_R3_from_R1_5
10.255.71.199   10.255.70.186   Up         163450       11114554 to_R3_from_R1_6
10.255.71.199   10.255.70.186   Up         132785        9029356 to_R3_from_R1_7
Total 9 displayed, Up 9, Down 0
```

What It Means This sample output shows that there are nine LSPs transiting R2. All are up and passing varying amounts of traffic.

Router Configurations for the Aggregated Interfaces Network

Purpose The configurations in this section are for the five routers in the example network illustrated in Figure 11 on page 92.

Action To display the configuration of a router, use the following JUNOS CLI operational mode command:

user@host> **show configuration | no-more**

Sample Output 1

```
user@R0> show configuration | no-more
[...Output truncated...]
interfaces {
    ge-1/2/0 {
        unit 0 {
            family inet {
                address 10.35.1.1/30;
            }
            family iso;
            family mpls;
        }
    }
}
routing-options {
    autonomous-system 69;
}
```

```
protocols {
    }
    mpls {
        traffic-engineering bgp-igp-both-ribs;
        interface all;
    }
    bgp {
        group int {
            type internal;
            local-address 10.255.71.197;
            family inet {
                any;
            }
            family inet-vpn {
                any;
            }
            neighbor 10.255.70.79;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface lo0.0;
                passive
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
    ldp {
        interface all;
        interface lo0.0
            passive
       }
    }
}
policy-options {
    policy-statement all_routes {
        then accept;
    }
}
```

Sample Output 2

```
user@R1> show configuration | no-more
[...Output truncated...]
chassis {
    redundancy {
        graceful-switchover {
            enable;
        }
    }
    aggregated-devices {
        sonet {
            device-count 1;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 10.35.1.2/30;
            }
```

```
                family iso;
                family mpls;
            }
        }
        so-7/2/0 {
            sonet-options {
                aggregate as0;
            }
        }
        so-7/2/1 {
            sonet-options {
                aggregate as0;
            }
        }
        as0 {
            aggregated-sonet-options {
                minimum-links 1;
            }
            unit 0 {
                family inet {
                    address 10.35.200.1/30;
                }
                family iso;
                family mpls;
            }
        }
    }
    routing-options {
        autonomous-system 69;
        forwarding-table {
            export pplb;
        }
    }
    protocols {
        rsvp {
            interface all;
        }
        mpls {
            traffic-engineering bgp-igp-both-ribs;
            label-switched-path to_R3_from_R1 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_1 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_2 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_3 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_4 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_5 {
                to 10.255.71.199;
                ldp-tunneling;
            }
```

```
            label-switched-path to_R3_from_R1_6 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            label-switched-path to_R3_from_R1_7 {
                to 10.255.71.199;
                ldp-tunneling;
            }
            interface all;
        }
        ospf {
            traffic-engineering;
            area 0.0.0.0 {
                interface lo0.0;
                    passive
                interface all;
                interface fxp0.0 {
                    disable;
                }
            }
        }
        ldp {
            interface ge-0/0/0.0;
            interface lo0.0;
        }
    }
    policy-options {
        policy-statement pplb {
            then {
                load-balance per-packet;
            }
        }
    }
```

Sample Output 3

```
user@R2> show configuration | no-more
[...Output truncated...]
chassis {
    redundancy {
        graceful-switchover {
            enable;
        }
    }
    aggregated-devices {
        ethernet {
            device-count 1;
        }
        sonet {
            device-count 3;
        }
    }
}
interfaces {
    ge-1/1/0 {
        gigether-options {
            802.3ad {
                ae0;
            }
        }
    }
    so-4/1/0 {
        sonet-options {
            aggregate as0;
        }
```

```
        }
        so-4/1/1 {
            sonet-options {
                aggregate as0;
            }
        }
        ge-4/3/1 {
            gigether-options {
                802.3ad {
                    ae0;
                }
            }
        }
        ae0 {
            aggregated-ether-options {
                minimum-links 1;
            }
            unit 0 {
                family inet {
                    address 10.35.200.5/30;
                }
                family iso;
                family mpls;
            }
        }
        as0 {
            aggregated-sonet-options {
                minimum-links 1;
            }
            unit 0 {
                family inet {
                    address 10.35.200.2/30;
                }
                family iso;
                family mpls;
            }
        }
    }
    forwarding-options {
        hash-key {
            family mpls {
                label-1;
                label-2;
                label-3;
            }
        }
    }
    routing-options {
        autonomous-system 69;
        forwarding-table {
            export pplb;
        }
    }
    protocols {
        rsvp {
            interface all;
        }
        mpls {
            traffic-engineering bgp-igp-both-ribs;
            interface all;
        }
        ospf {
            traffic-engineering;
```

```
        area 0.0.0.0 {
            interface lo0.0;
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
}
policy-options {
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
```

Sample Output 4

```
user@R3> show configuration | no-more
[...Output truncated...]
chassis {
    redundancy {
        graceful-switchover {
            enable;
        }
    }
    aggregated-devices {
        ethernet {
            device-count 1;
        }
        sonet {
            device-count 2;
        }
    }
}
interfaces {
    ge-1/3/0 {
        gigether-options {
            802.3ad {
                ae0;
            }
        }
    }
    ge-4/0/1 {
        gigether-options {
            802.3ad {
                ae0;
            }
        }
    }
    ge-7/0/0 {
        unit 0 {
            family inet {
                address 10.35.1.53/30;
            }
            family iso;
            family mpls;
        }
    }
    ae0 {
        aggregated-ether-options {
            minimum-links 1;
        }
        unit 0 {
```

```
                family inet {
                    address 10.35.200.6/30;
                }
                family iso;
                family mpls;
            }
        }
    }
    routing-options {
        autonomous-system 69;
    }
    protocols {
        rsvp {
            interface all;
        }
        mpls {
            traffic-engineering bgp-igp-both-ribs;
            label-switched-path to_R1_from_R3 {
                to 10.255.70.186;
                ldp-tunneling;
            }
            interface all;
        }
        ospf {
            traffic-engineering;
            area 0.0.0.0 {
                interface lo0.0;
                interface all;
                interface fxp0.0 {
                    disable;
                }
            }
        }
        ldp {
            interface ge-7/0/0.0;
            interface lo0.0;
        }
    }
```

Sample Output 5

```
user@R4> show configuration | no-more
[...Output truncated...]
interfaces {
    ge-0/3/0 {
        unit 0 {
            family inet {
                address 10.35.1.54/30;
            }
            family iso;
            family mpls;
        }
    }
}
routing-options {
    autonomous-system 69;
}
protocols {
    mpls {
        traffic-engineering bgp-igp-both-ribs;
        interface all;
    }
    bgp {
        group int {
            type internal;
```

```
                local-address 10.255.70.79;
                family inet {
                    any;
                }
                family inet-vpn {
                    any;
                }
            }
        }
        ospf {
            traffic-engineering;
            area 0.0.0.0 {
                interface lo0.0;
                interface all;
                interface fxp0.0 {
                    disable;
                }
            }
        }
        ldp {
            interface all;
        }
    }
```

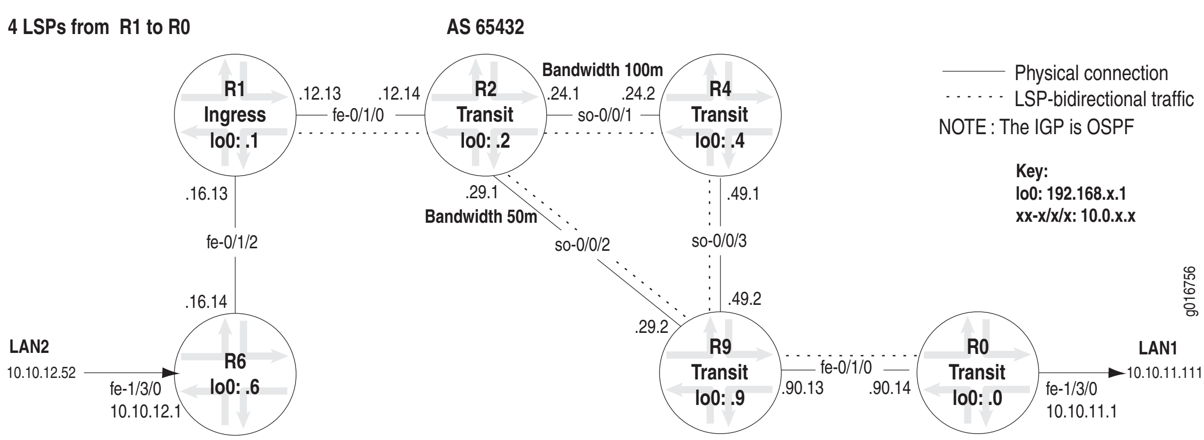
What It Means Sample Outputs 1 through 5 show the configuration of all routers in the example network shown in Figure 11 on page 92.

Example: Load-Balancing a Network Using INET in the Hash Key

Purpose The IPv4 address family (INET) provides additional information to identify traffic flows and balance traffic more evenly. You configure the INET or port data on an ingress router. Configuring port data is useful if you are using TCP or UDP. However, it may not be useful to include port data when you are using protocols that are not associated with a Layer 4 port, for example, Layer 2 VPNs, GRE tunneling, or ICMP.

The following network example shows the process for verifying the operation of the hash key configuration of port data as described in “Configuring the IPv4 Address Family to Load-Balance LSP Traffic” on page 89.

Figure 12: INET Network Topology



The network topology in Figure 11 on page 92 illustrates a router-only network with SONET and Ethernet interfaces that consists of the following components:

- A full-mesh interior BGP (IBGP) topology, using AS 65432
- MPLS and RSVP enabled on all routers
- A send-statics policy on routers R1 and R0 that allows a new route to be advertised into the network
- Four unidirectional LSPs between R1 and R0, and one reverse direction LSP between R0 and R1, which allows for bidirectional traffic
- Load balancing configured on the ingress router R1
- The hash key using port data configured on R1
- Bandwidth configured on the SONET interfaces on R2

In addition, the example network uses Open Shortest Path First (OSPF) as the interior gateway protocol (IGP) with OSPF area 0.0.0.0. An IGP is required for the Constrained Shortest Path First (CSPF) LSP, which is the default for the JUNOS software. Also, the example network uses a policy to create BGP traffic.

The following information is included in this example:

- Verifying the Operation of INET Load Balancing on page 104
- Router Configurations for the INET Load-Balanced Network on page 107

Verifying the Operation of INET Load Balancing

Action On the ingress router, to verify the operation of the hash key, enter the following JUNOS CLI operational mode commands:

user@host> **show configuration**
user@host# **show route forwarding-table destination** *destination*

On the transit router, to verify the operation of the hash key, enter the following JUNOS CLI operational mode commands:

user@host> **show route**
user@host> **monitor interface traffic**
user@host> **show mpls lsp statistics**

Sample Output The following sample output is for ingress router R1:

```
user@R1> show configuration forwarding-options
hash-key {
    family inet { #Port data configuration
        layer-3;
        layer-4;
    }
}
```

```
user@R1> show configuration routing-options
static {
[...Output truncated...]
autonomous-system 65432;
forwarding-table {
    export lbpp; #Load balancing policy applied
}

user@R1> show configuration policy-options
policy-statement lbpp { #Load balancing policy defined
    then {
        load-balance per-packet;
    }
}
[...Output truncated...]
```

What It Means The sample output from ingress router R1 for the three show configuration commands (forwarding-options, routing-options, and policy-options) shows that load balancing is correctly configured for the INET hash key and the load-balancing policy (lbpp).

Sample Output The following sample output is for ingress router R1:

```
user@R1> show route forwarding-table destination 10.0.90.14
Routing table: inet
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
10.0.90.14/32      user     0                    ulst 262144     2
                              10.0.12.14        Push 100688       fe-0/1/0.0
                              10.0.12.14        Push 100656       fe-0/1/0.0
                              10.0.12.14        Push 100672       fe-0/1/0.0
                              10.0.12.14        Push 100704       fe-0/1/0.0
```

What It Means The sample output from ingress router R1 for the show route forwarding-table destination command shows unilist (ulst) in the Type field, indicating that load balancing is working. In this case, the Type field shows the operation performed on packets. The push operation adds a new label to the top of the packet before the packets are pushed out of interface fe-0/1/0.0.

Sample Output The following sample output is for transit router R2:

```
user@R2> show route 10.0.90.14

inet.0: 25 destinations, 27 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.90.12/30      *[OSPF/10] 03:06:04, metric 3
                      via so-0/0/1.0
                    > via so-0/0/2.0
```

What It Means The sample output from transit router R2 for the show route command shows two OSPF routes to the destination interface on egress router R0. Even though the route with the greater than sign (>) is the selected route, traffic will be balanced across both interfaces, as shown in the output for the following show route forwarding-table and monitor traffic commands.

Sample Output The following sample output is for transit router R2:

```
user@R2> show route forwarding-table destination 10.0.90.14
Routing table: inet
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
10.0.90.12/30      user     0                     ulst 262144     6
                                                 ucst   345     5 so-0/0/1.0
                                                 ucst   339     2 so-0/0/2.0
```

What It Means The sample output from transit router R2 for the show route forwarding-table destination command shows unilist (ulst) in the Type field, indicating that load balancing is working. A packet sent to this next hop (R2) goes to any next hop in the unicast (ucst) list, so-0/0/1.0 and so-0/0/2.0.

Sample Output The following sample output is for transit router R2:

```
user@R2> monitor interface traffic
R2                                  Seconds: 123                Time: 21:28:29

Interface    Link  Input packets        (pps)     Output packets        (pps)
 so-0/0/0      Up              0          (0)                  0          (0)
 so-0/0/1      Up             95          (0)              50012          (1)
 so-0/0/2      Up         100132         (19)              50217          (0)
 so-0/0/3      Up              0          (0)                  0          (0)
 fe-0/1/0      Up         100127         (17)             100128          (1)
 fe-0/1/1      Up              0          (0)                  0          (0)
 fe-0/1/2      Up              0          (0)                  0          (0)
 fe-0/1/3      Up              0          (0)                  0          (0)
 [...Output truncated...]
```

What It Means The sample output from transit router R2 for the monitor interface traffic command shows that traffic is balanced across interfaces so-0/0/1.0 and so-0/0/2.0.

Sample Output The following sample output is for transit router R2:

```
user@R2> show mpls lsp statistics
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 5 sessions
To              From            State     Packets            Bytes LSPname
192.168.0.1     192.168.1.1     Up          24874          2188912 lsp1
192.168.0.1     192.168.1.1     Up          24471          2153448 lsp2
192.168.0.1     192.168.1.1     Up          25613          2253944 lsp3
192.168.0.1     192.168.1.1     Up          25042          2203696 lsp4
192.168.1.1     192.168.0.1     Up              0                0 r0-r1
Total 5 displayed, Up 5, Down 0
```

What It Means The sample output from transit router R2 for the show mpls lsp statistics command shows that traffic is balanced across the four LSPs (lsp1, lsp2, lsp3, and lsp4) transiting R2.

Router Configurations for the INET Load-Balanced Network

Purpose The configurations in this section are for the six routers in the example network illustrated in Figure 12 on page 103.

Action To display a router configuration, use the following JUNOS CLI operational mode command:

user@host> **show configuration | no-more**

Sample Output 1 The following sample output is for edge router R6:

```
user@R6> show configuration | no-more
interfaces {
    fe-0/1/2 { #Interface connected to R1
        unit 0 {
            family inet {
                address 10.0.16.14/30;
            }
            family mpls; # #MPLS enabled on relevant interfaces
        }
    }
    fe-1/3/0 {
        unit 0 {
            family inet {
                address 10.10.12.1/24;
            }
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.148/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.6.1/32;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
    }
    router-id 192.168.6.1; #Manually configured RID
    autonomous-system 65432; #Full mesh IBGP
}
protocols {
    rsvp {
        interface fe-0/1/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface fe-0/1/2.0;
        interface fxp0.0 {
```

```
                disable;
            }
        }
        bgp {
            group internal {
                type internal;
                local-address 192.168.6.1;
                neighbor 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.9.1;
                neighbor 192.168.0.1;
            }
        }
        ospf { #IGP enabled
            traffic-engineering;
            area 0.0.0.0 {
                interface fe-0/1/2.0;
                interface fe-1/3/0.0;
                interface lo0.0 {
                    passive; #Ensures protocols do not run over this interface
                }
            }
        }
    }
```

Sample Output 2 The following sample output is for ingress router R1:

```
user@R1> show configuration | no-more
interfaces {
    fe-0/1/0 { #Connected to R2
        unit 0 {
            family inet {
                address 10.0.12.13/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    fe-0/1/2 { #Connected to R6
        unit 0 {
            family inet {
                address 10.0.16.13/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.143/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
        }
    }
}
forwarding-options {
```

```
    hash-key {
        family inet { INET/port data
            layer-3;
            layer-4;
        }
    }
}
routing-options {
    static {
        [...Output truncated...
        }
        route 100.100.1.0/24 reject; #Static route for send-statics policy
    }
    router-id 192.168.1.1; #Manually configured RID
    autonomous-system 65432; #Full mesh IBGP
    forwarding-table {
        export lbpp; #Routes exported to forwarding table
    }
}
protocols {
    rsvp {
        interface fe-0/1/0.0;
        interface fe-0/1/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path lsp1 { #First LSP
            to 192.168.0.1; # Destination of the LSP
            install 10.0.90.14/32 active; # The prefix is installed in the
            primary via-r4;               # inet.0 routing table
        }
        label-switched-path lsp2 {
            to 192.168.0.1;
            install 10.0.90.14/32 active;
            primary via-r2;
        }
        label-switched-path lsp3 {
            to 192.168.0.1;
            install 10.0.90.14/32 active;
            primary via-r2;
        }
        label-switched-path lsp4 {
            to 192.168.0.1;
            install 10.0.90.14/32 active;
            primary via-r4;
        }
        path via-r2 { #Primary path to spread traffic across interfaces
            10.0.29.2 loose;
        }
        path via-r4 {
            10.0.24.2 loose;
        }
        interface fe-0/1/0.0;
        interface fe-0/1/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        export send-statics; #Allows advertising of a new route
        group internal {
```

```
                type internal;
                local-address 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.9.1;
                neighbor 192.168.6.1;
                neighbor 192.168.0.1;
            }
        }
        ospf { #IGP enabled
            traffic-engineering;
            area 0.0.0.0 {
                interface fe-0/1/0.0;
                interface fe-0/1/2.0;
                interface lo0.0 {
                    passive;
                }
            }
        }
    }
    policy-options {
        policy-statement lbpp { #Load balancing policy
            then {
                load-balance per-packet;
            }
        }
        policy-statement send-statics { #Static route policy
            term statics {
                from {
                    route-filter 100.100.1.0/24 exact;
                }
                then accept;
            }
        }
    }
```

Sample Output 3 The following sample output is for transit router R2:

```
user@R2> show configuration | no-more
interfaces {
    so-0/0/1 { #Connected to R4
        unit 0 {
            bandwidth 100m; #Bandwidth to ensure equal-cost paths
            family inet {
                address 10.0.24.1/30;
            }
            family mpls; #MPLS enabled on relevant interfaces
        }
    }
    so-0/0/2 { #Connected to R9
        unit 0 {
            bandwidth 50m; #Bandwidth to ensure equal-cost paths
            family inet {
                address 10.0.29.1/30;
            }
            family mpls;
        }
    }
    fe-0/1/0 { Connected to R1
        unit 0 {
            family inet {
```

```
                address 10.0.12.14/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.144/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.2.1/32;
            }
        }
    }
}
forwarding-options {
    hash-key {
        family mpls { #MPLS labels configuration
            label-1;
            label-2;
            payload {
                ip;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
    }
    router-id 192.168.2.1;
    autonomous-system 65432;
    forwarding-table {
        export lbpp; #Routes exported into forwarding table
    }
}
protocols {
    rsvp {
        interface so-0/0/1.0;
        interface fe-0/1/0.0;
        interface so-0/0/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface fe-0/1/0.0;
        interface so-0/0/1.0;
        interface so-0/0/2.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.2.1;
            neighbor 192.168.1.1;
```

```
                neighbor 192.168.4.1;
                neighbor 192.168.9.1;
                neighbor 192.168.6.1;
                neighbor 192.168.0.1;
            }
        }
        ospf {
            traffic-engineering;
            area 0.0.0.0 {
                interface fe-0/1/0.0;
                interface so-0/0/1.0;
                interface so-0/0/2.0;
                interface lo0.0 {
                    passive;
                }
            }
        }
    }
    policy-options {
        policy-statement lbpp { #Load balancing policy exported in forwarding table
            then {
                load-balance per-packet;
            }
        }
    }
```

Sample Output 4 The following sample output is for transit router R4:

```
user@R4> show configuration | no-more
interfaces {
    so-0/0/1 {
        unit 0 {
            family inet {
                address 10.0.24.2/30;
            }
            family mpls;
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.0.49.1/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.146/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.4.1/32;
            }
        }
    }
}
routing-options {
```

```
    static {
        [...Output truncated...]
    }
    router-id 192.168.4.1;
    autonomous-system 65432;
}
protocols {
    rsvp {
        interface so-0/0/1.0;
        interface so-0/0/3.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface so-0/0/1.0;
        interface so-0/0/3.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.4.1;
            neighbor 192.168.1.1;
            neighbor 192.168.2.1;
            neighbor 192.168.9.1;
            neighbor 192.168.6.1;
            neighbor 192.168.0.1;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface so-0/0/1.0;
            interface so-0/0/3.0;
            interface lo0.0 {
                passive;
            }
        }
    }
}
```

Sample Output 5 The following sample output is for transit router R9:

```
user@R9> show configuration | no-more
interfaces {
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.0.29.2/30;
            }
            family mpls;
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.0.49.2/30;
            }
```

```
                family mpls;
            }
        }
        fe-0/1/0 {
            unit 0 {
                family inet {
                    address 10.0.90.13/30;
                }
                family mpls;
            }
        }
        fxp0 {
            unit 0 {
                family inet {
                    address 192.168.69.206/21;
                }
            }
        }
        lo0 {
            unit 0 {
                family inet {
                    address 192.168.9.1/32;
                }
            }
        }
    }
    routing-options {
        static {
            [...Output truncated...]
        }
        router-id 192.168.9.1;
        autonomous-system 65432;
    }
    protocols {
        rsvp {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface fxp0.0 {
                disable;
            }
        }
        mpls {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface fxp0.0 {
                disable;
            }
        }
        bgp {
            group internal {
                type internal;
                local-address 192.168.9.1;
                neighbor 192.168.1.1;
                neighbor 192.168.2.1;
                neighbor 192.168.4.1;
                neighbor 192.168.0.1;
                neighbor 192.168.6.1;
            }
        }
        ospf {
            traffic-engineering;
```

```
        area 0.0.0.0 {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface lo0.0 {
                passive;
            }
        }
    }
}
```

Sample Output 6 The following sample output is for egress router R0:

```
user@R0> show configuration | no-more
interfaces {
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.0.29.2/30;
            }
            family mpls;
        }
    }
    so-0/0/3 {
        unit 0 {
            family inet {
                address 10.0.49.2/30;
            }
            family mpls;
        }
    }
    fe-0/1/0 {
        unit 0 {
            family inet {
                address 10.0.90.13/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.69.206/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.9.1/32;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
    }
    router-id 192.168.9.1;
    autonomous-system 65432;
}
protocols {
```

```
    rsvp {
        interface so-0/0/2.0;
        interface so-0/0/3.0;
        interface fe-0/1/0.0;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        interface so-0/0/2.0;
        interface so-0/0/3.0;
        interface fe-0/1/0.0;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group internal {
            type internal;
            local-address 192.168.9.1;
            neighbor 192.168.1.1;
            neighbor 192.168.2.1;
            neighbor 192.168.4.1;
            neighbor 192.168.0.1;
            neighbor 192.168.6.1;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface so-0/0/2.0;
            interface so-0/0/3.0;
            interface fe-0/1/0.0;
            interface lo0.0 {
                passive;
            }
        }
    }
}
```

Using Bandwidth to Unevenly Load-Balance RSVP LSPs

Purpose With RSVP LSPs, load-balancing LSP traffic using bandwidth allows uneven load balancing across multiple external links that have varying amounts of available bandwidth. When you use bandwidth to load-balance an RSVP LSP, the distribution of traffic is proportional to the bandwidth configuration of each LSP. You configure load balancing at the [edit protocols rsvp] hierarchy level on the ingress router.

For uneven load balancing using bandwidth to work, you must have at least two equal-cost LSPs toward the same egress router and at least one of the LSPs must have a bandwidth value configured at the [edit protocols mpls label-switched-path *lsp-path-name*] hierarchy level. If no LSPs have bandwidth configured, equal distribution load balancing is performed. If only some LSPs have bandwidth configured, the LSPs without any bandwidth configured do not receive any traffic.

Keep the following information in mind when you use the load-balance statement at the [edit protocols rsvp] hierarchy level:

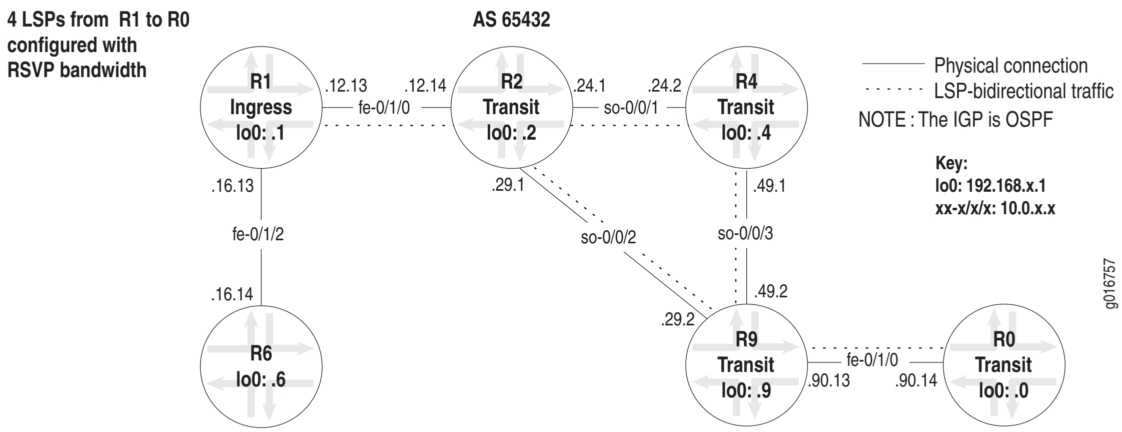
- The behavior of currently running LSPs is not altered. To force the currently running LSPs to use the new behavior, issue the clear mpls lsp command.
- The load-balance statement at the [edit protocols rsvp] hierarchy level only applies to ingress LSPs that have a policy with the load-balancing per-packet statement configured.
- For Differentiated Services-aware traffic-engineered LSPs, the bandwidth of an LSP is calculated by summing the bandwidth of all of the class types.

Before you can use bandwidth to unevenly load-balance LSP traffic, you must have the following configured on the ingress router:

- A policy with the load-balance per-packet statement at the [edit policy-options] hierarchy level and that policy applied as an export policy at the [edit forwarding-options] hierarchy level. For more information about configuring load balancing, see "Configuring and Verifying Load Balancing" on page 70.
- Bandwidth configured for each LSP at the [edit protocols mpls label-switched-path *lsp-path-name*] hierarchy level. For more information on configuring LSP bandwidth, see the *JUNOS MPLS Applications Configuration Guide*.

Figure 13 illustrates a network configured with RSVP bandwidth.

Figure 13: RSVP Bandwidth Network



The network topology in Figure 13 illustrates a router-only network with SONET and Ethernet interfaces that consists of the following components:

- A full-mesh IBGP topology, using AS 65432
- MPLS and RSVP enabled on all routers
- A send-statics policy on routers R1 and R0 that allows a new route to be advertised into the network
- Four unidirectional LSPs configured with uneven bandwidth between R1 and R0
- One reverse direction LSP between R0 and R1, which allows for bidirectional traffic
- Load balancing configured on the ingress router R1
- RSVP bandwidth configured on the ingress router R1

In addition, the example network uses OSPF as the IGP with OSPF area 0.0.0.0. An IGP is required for the CSPF LSP, which is the default for the JUNOS software. Also, the example network uses a policy to create BGP traffic.

The following information is included in this example:

1. Configure Bandwidth to Unevenly Load-Balance Traffic on page 119
2. Verify the Operation of Uneven Bandwidth Load Balancing on page 120
3. Router Configurations for Bandwidth Load Balancing on page 122

Step 1: Configure Bandwidth to Unevenly Load-Balance Traffic

Purpose Configuring bandwidth to unevenly load-balance traffic is performed in three stages. The first stage enables a load-balancing policy, the second stage configures the LPS bandwidth, and the third stage enables RSVP load balancing.

Action To configure bandwidth to unevenly load-balance RSVP LSPs, follow these steps:

1. Ensure that you have load balancing configured: see "Configuring and Verifying Load Balancing" on page 70.

2. Configure LSP bandwidth. In configuration mode, go to the following hierarchy level:

   [edit]
   user@host# **edit protocols mpls**

3. Configure the LSP bandwidth:

   [edit protocols mpls]
   user@host# **set label-switched-path** *lsp-path-name* **bandwidth** *bps*

4. Verify the configuration:

   [edit protocols mpls]
   user@host# **show**

5. Configure RSVP bandwidth. Go to the following hierarchy level:

   [edit]
   user@host# **edit protocols rsvp**

6. Configure the bandwidth statement:

   [edit protocols rsvp]
   user@host# **set load-balance bandwidth**

7. Verify and commit the configuration:

   user@host# **show**
   user@host# **commit**

Sample Output

```
user@R1> edit
Entering configuration mode

[edit]
user@R1# edit protocols mpls

[edit protocols mpls]
user@R1# set label-switched-path lsp1 bandwidth 10m

[edit protocols mpls]
user@R1# show
label-switched-path lsp1 {
    to 192.168.0.1;
    install 10.0.90.14/32 active;
    bandwidth 10m;
    primary via-r4;
```

```
[edit protocols mpls]
user@R1# top

[edit]
user@R1# edit protocols rsvp

[edit protocols rsvp]
user@R1# set load-balance bandwidth

[edit protocols rsvp]
user@R1# show
load-balance bandwidth;
interface fe-0/1/2.0;
interface fxp0.0 {
    disable;
}

[edit protocols rsvp]
user@R1# commit
commit complete
```

What It Means The sample output shows the configuration of LSP bandwidth and RSVP bandwidth on ingress router R1. The sample output shows only one LSP configured with bandwidth, however, for RSVP bandwidth to work, you must have at least two equal-cost LSPs toward the same egress router and at least one of the LSPs must have a bandwidth value configured. If no LSPs have bandwidth configured, equal-distribution load balancing is performed. If only some LSPs have bandwidth configured, the LSPs without any bandwidth configured do not receive any traffic.

Step 2: Verify the Operation of Uneven Bandwidth Load Balancing

Purpose When a router is performing unequal-cost load balancing between LSPs paths, the show route detail command displays a balance field associated with each next hop being used.

Action To verify that an RSVP LSP is unevenly load-balanced, use the following JUNOS CLI operational mode commands:

user@host> **show route protocol rsvp detail**
user@host> **show mpls lsp statistics**

Sample Output

```
user@R1> show route protocol rsvp detail

inet.0: 25 destinations, 25 routes (25 active, 0 holddown, 0 hidden)
10.0.90.14/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 7
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 10%
                Label-switched-path lsp1
                Label operation: Push 100768
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 20%
                Label-switched-path lsp2
                Label operation: Push 100736
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 30% ,selected
                Label-switched-path lsp3
                Label operation: Push 100752
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 40%
```

```
                **Label-switched-path lsp4**
                Label operation: Push 100784
                State: <Active Int>
                Local AS: 65432
                Age: 8:03       Metric: 4
                Task: RSVP
                Announcement bits (2): 0-KRT 4-Resolve tree 1
                AS path: I

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

192.168.0.1/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 7
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 10%
                Label-switched-path lsp1
                Label operation: Push 100768
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 20%
                Label-switched-path lsp2
                Label operation: Push 100736
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 30%
                Label-switched-path lsp3
                Label operation: Push 100752
                Next hop: 10.0.12.14 via fe-0/1/0.0 weight 0x1 balance 40%, selected
                Label-switched-path lsp4
                Label operation: Push 100784
                State: <Active Int>
                Local AS: 65432
                Age: 8:03       Metric: 4
                Task: RSVP
                Announcement bits (1): 1-Resolve tree 1
                AS path: I
```

```
user@R1> show mpls lsp statistics
Ingress LSP: 4 sessions
To               From             State     Packets           Bytes LSPname
192.168.0.1      192.168.1.1      Up          10067          845628 lsp1
192.168.0.1      192.168.1.1      Up          20026         1682184 lsp2
192.168.0.1      192.168.1.1      Up          29796         2502864 lsp3
192.168.0.1      192.168.1.1      Up          40111         3369324 lsp4
Total 4 displayed, Up 4, Down 0

Egress LSP: 1 sessions
To               From             State     Packets           Bytes LSPname
192.168.1.1      192.168.0.1      Up             NA              NA r0-r1
Total 1 displayed, Up 1, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

What It Means The sample output from ingress router R1 shows that traffic is distributed according to the LSP bandwidth configuration, as indicated by the Balance: xx% field. For example, lsp1 has 10 Mbps of bandwidth configured, as reflected in the Balance: 10% field.

Router Configurations for Bandwidth Load Balancing

Purpose The configuration in this section is for ingress router R1 in the example network illustrated in Figure 13 on page 118. The configuration for the other five routers in the network are the same as those found in "Router Configurations for the Load-Balanced MPLS Network" on page 76.

Action To display a router configuration, use the following JUNOS CLI operational mode command:

user@host> **show configuration | no-more**

Sample Output

```
user@R1> show configuration | no-more
[....Output truncated...]
interfaces {
    fe-0/1/0 {
        unit 0 {
            family inet {
                address 10.0.12.13/30;
            }
            family mpls;
        }
    }
    fe-0/1/2 {
        unit 0 {
            family inet {
                address 10.0.16.13/30;
            }
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.143/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
        }
    }
}
routing-options {
    static {
        [...Output truncated...]
        }
        route 100.100.1.0/24 reject;
    }
    router-id 192.168.1.1;
    autonomous-system 65432;
    forwarding-table {
        export lbpp;
    }
}
protocols {
    rsvp {
        load-balance bandwidth; #RSVP bandwidth load balancing
```

```
    interface fe-0/1/0.0;
    interface fe-0/1/2.0;
    interface fxp0.0 {
        disable;
    }
}
mpls {
    label-switched-path lsp1 {
        to 192.168.0.1;
        install 10.0.90.14/32 active;
        bandwidth 10m; #Bandwidth configured for each LSP
        primary via-r4;
    }
    label-switched-path lsp2 {
        to 192.168.0.1;
        install 10.0.90.14/32 active;
        bandwidth 20m; #Bandwidth configured for each LSP
        primary via-r2;
    }
    label-switched-path lsp3 {
        to 192.168.0.1;
        install 10.0.90.14/32 active;
        bandwidth 30m; #Bandwidth configured for each LSP
        primary via-r2;
    }
    label-switched-path lsp4 {
        to 192.168.0.1;
        install 10.0.90.14/32 active;
        bandwidth 40m; #Bandwidth configured for each LSP
        primary via-r4;
    }
    path via-r2 {
        10.0.29.2 loose;
    }
    path via-r4 {
        10.0.24.2 loose;
    }
    interface fe-0/1/0.0;
    interface fe-0/1/2.0;
    interface fxp0.0 {
        disable;
    }
}
bgp {
    export send-statics;
    group internal {
        type internal;
        local-address 192.168.1.1;
        neighbor 192.168.2.1;
        neighbor 192.168.4.1;
        neighbor 192.168.9.1;
        neighbor 192.168.6.1;
        neighbor 192.168.0.1;
    }
}
ospf {
    traffic-engineering;
    area 0.0.0.0 {
        interface fe-0/1/0.0;
        interface fe-0/1/2.0;
        interface lo0.0 {
            passive;
        }
```

```
        }
    }
}
policy-options {
    policy-statement lbpp { Load balancing policy
        then {
            load-balance per-packet;
        }
    }
    policy-statement send-statics {
        term statics {
            from {
                route-filter 100.100.1.0/24 exact;
            }
            then accept;
        }
    }
}
```

What It Means The sample output shows the configuration for the ingress router R1 in the example network illustrated in Figure 13 on page 118. The configuration for the other five routers in the network is the same as those found in "Router Configurations for the Load-Balanced MPLS Network" on page 76.

Traffic Flows Before Load Balancing

Purpose The following sample output illustrates the details to look for when you issue different show commands to check if traffic is balanced. The following output is before load balancing is configured and is taken from transit router R2 in the network shown in Figure 10 on page 75.

Action To check the distribution of traffic across interfaces and LSPs, use the following CLI operational mode commands:

user@host> **show route | find mpls**
user@host> **monitor interface traffic**
user@host> **show mpls lsp statistics**

Sample Output 1

```
user@R2> show route | find mpls

mpls.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0                  *[MPLS/0] 1d 00:12:08, metric 1
                      Receive
1                  *[MPLS/0] 1d 00:12:08, metric 1
                      Receive
2                  *[MPLS/0] 1d 00:12:08, metric 1
                      Receive
100112             *[RSVP/7] 13:10:36, metric 1
                    > via so-0/0/1.0, label-switched-path lsp1
100128             *[RSVP/7] 13:01:08, metric 1
                    > via so-0/0/1.0, label-switched-path lsp4
100144             *[RSVP/7] 00:26:49, metric 1
                    > to 10.0.12.13 via fe-0/1/0.0, label-switched-path r0-r6
100160             *[RSVP/7] 00:23:25, metric 1
                    > via so-0/0/2.0, label-switched-path lsp2
100176             *[RSVP/7] 00:23:25, metric 1
                    > via so-0/0/2.0, label-switched-path lsp3
```

Sample Output 2

```
user@R2> monitor interface traffic

R2                                Seconds: 89                Time: 14:33:09

Interface    Link  Input packets        (pps)     Output packets        (pps)
 so-0/0/0     Up               0          (0)                  0          (0)
 so-0/0/1     Up              90          (1)                 91          (1)
 so-0/0/2     Up             118          (1)             100122          (0)
 so-0/0/3     Up               0          (0)                  0          (0)
 fe-0/1/0     Up          100119          (0)                115          (0)
 fe-0/1/1     Up               0          (0)                  0          (0)
 fe-0/1/2     Up               0          (0)                  0          (0)
 fe-0/1/3     Up               0          (0)                  0          (0)
 [...Output truncated...]
```

Sample Output 3

```
user@R2> show mpls lsp statistics
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 5 sessions
To              From            State     Packets            Bytes LSPname
192.168.0.1     192.168.6.1     Up              0                0 lsp2
192.168.0.1     192.168.6.1     Up         112026         22853304 lsp1
192.168.0.1     192.168.6.1     Up              0                0 lsp3
192.168.0.1     192.168.6.1     Up              0                0 lsp4
192.168.6.1     192.168.0.1     Up              0                0 r0-r6
Total 5 displayed, Up 5, Down 0
```

What It Means

Sample Outputs 1 through 3 from transit router R2 show that traffic is not balanced across LSPs or interfaces.

Sample Output 1 for the show route command shows that all LSPs have the same metric (1) to the destination, even though they are traversing different interfaces. lsp1 and lsp4 are using so-0/0/1, while lsp2 and lsp3 are using so-0/0/2.

Sample Output 2 for the monitor interface traffic command shows that traffic is not evenly balanced across interfaces so-0/0/1 and so-0/0/2. Almost all traffic is going out so-0/0/2.

Sample Output 3 for the show mpls lsp statistics command shows that traffic across LSPs is not balanced. All traffic is going over lsp1.

Related Information

For additional information about MPLS fast reroute and MPLS protection methods, see the following:

- *JUNOS Feature Guide*
- *JUNOS MPLS Applications Configuration Guide*
- Semeria, Chuck. *RSVP Signaling Extensions for MPLS Traffic Engineering.* White paper. 2002
- Semeria, Chuck. *IP Dependability: Network Link and Node Protection*. White paper. 2002
- RFC 4090, *Fast Reroute Extensions to RSVP-TE for LSP Tunnels*

The JUNOS software uses the load-balancing function across different protocols and features. For information about other types of load balancing, see the following:

- "Option: Optimizing VPLS Traffic Flows", *JUNOS Feature Guide*
- "Protocol-Independent Load Balancing for Layer 3 VPNs", *JUNOS VPNs Configuration Guide*
- "Load Balancing Among Multiple Monitoring Interfaces", *JUNOS Services Interfaces Configuration Guide*