Chapter 6

Troubleshooting Fast Reroute

This case study describes a problem establishing Fast Reroute (FRR) link protection in a Multiprotocol Label Switching (MPLS)-based VPN. Specifically, FRR requires a load balancing policy for the correct installation of routes in the forwarding table and fast local repair. The principles and solution used in this case study apply to all forms of local protection. For an overview of local protection, see "Local Protection in an MPLS Network" on page 25.

The chapter includes a brief summary of the FRR problem within the context of an MPLS-based VPN, an example network scenario, and commands to troubleshoot and resolve the problem. (See Table 9.)

The troubleshooting process described in this case study should not be followed rigidly; it is a basis from which you can develop your own process to suit your particular situation.

Table 9: Troubleshooting Fast Reroute Checklist

Troubleshooting Fast Reroute Tasks	Command or Action
Fast Reroute Problem Overview on page 131	
Symptom on page 132	Local repair is taking about one second) to complete, which is slow. show route forwarding-table extensive
Cause on page 132	The forwarding table does not include the necessary next-hops to support local repair.
Troubleshooting Commands on page 132	show configuration routing-instances *routing-instance-name* show bgp summary instance routing-*instance-name* show configuration protocols mpls show mpls lsp ingress show rsvp session ingress show rsvp session ingress detail show route table *table destination* detail show route forwarding-table vpn *vpn* destination *destination* extensive
Solution on page 137	Enable load-balancing and ensure that multiple next-hop forwarding table entries appear in the forwarding table for each destination. show configuration policy-options show configuration routing-options show route forwarding-table vpn *vpn* destination *destination* extensive

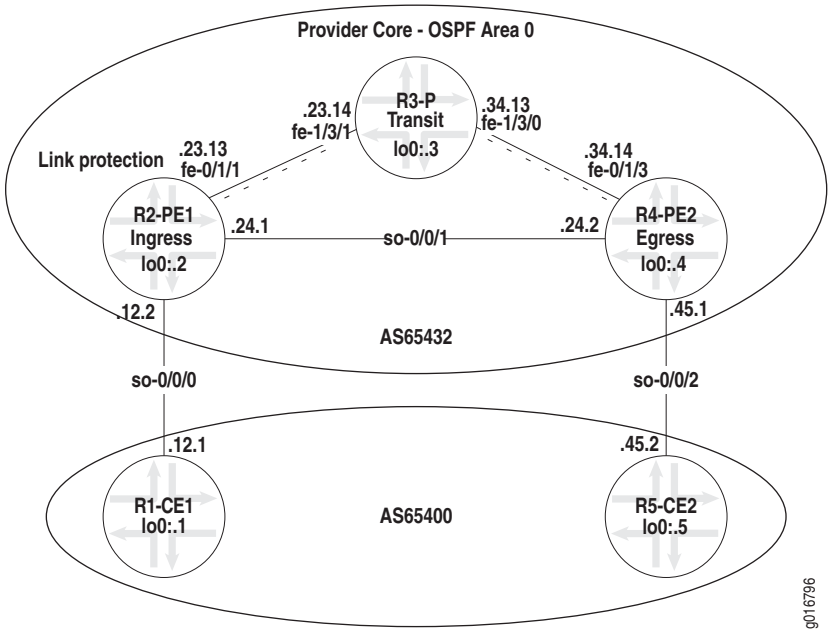
Troubleshooting Fast Reroute Tasks	Command or Action
Conclusion on page 138	A load balancing policy is required for link protection to work effectively. The principles are the same for the configuration of the fast reroute and the node-link-protection statements.
Router Configurations on page 138	show configuration | no-more

Fast Reroute Problem Overview

Incorrect configuration is a common mistake when trying to establish protection for an MPLS LSP. Protection with either fast reroute or link protection requires a per-packet load-balance policy exported at the [edit routing-options forwarding-table] hierarchy level. Correctly configured protection for an MPLS LSP results in two next-hop forwarding table entries per destination, either an incoming MPLS label or an IP destination. For information on configuring FRR, see "MPLS FRR Protection Overview" on page 3.

Figure 14 illustrates a network topology with link protection and load balancing enabled to ensure that routes are correctly placed in to the forwarding table.

Figure 14: Fast Reroute Problem Network



The network shown in Figure 14 illustrates an MPLS-based VPN with traffic protection and load balancing, consisting of the following:

- All physical interfaces addresses are from the 10.0.x.x/30 address space.
- All loopback addresses are from the 192.168.x.1/32 block.
- The IGP is a single-area (Area 0) OSPF.
- RSVP is deployed as the MPLS signaling protocol between PE routers.
- LSPs (r2-r4 and r4-r2) established between PE routers.
- MP-IBGP mesh between PE routers, loopback peering, and VPN-IPv4 NLRI.

- CE-PE link running EBGP.
- Full-mesh Layer 3 VPN between CE1 and CE2.
- Traffic protection for the link between the PE1 and P routers.
- Load balancing on PE1.

The overall goal of this network is to provide point-to-point connectivity between the two CE routers and traffic protection in the core of the network.

Symptom

In the network shown in Figure 14, the external symptom is that local repair is taking about one second to complete, which is slow. Use the show route forwarding-table vpn vpn-a destination command to check that the correct routes are included in the forwarding table. In the example output below, there is only one route installed in the forwarding table, when for fast local repair, there should be multiple next hops installed.

```
user@R2-PE1> show route forwarding-table vpn vpn-a destination 192.168.5.1
extensive
Routing table: vpn-a.inet [Index 2]
Internet:

Destination:  192.168.5.0/24
  Route type: user
  Route reference: 0                    Route interface-index: 0
  Flags: sent to PFE, prefix load balance
  Next-hop type: indirect               Index: 262142   Reference: 2
  Next-hop type: Push 100160
  Next-hop interface: so-0/0/1.0 #Only one next hop in the forwarding table.
```

Cause

Slow local repair is caused by the forwarding table not including the necessary next-hops to support local repair. The forwarding table shows only a single next-hop, when local repair requires additional next-hops for fast recovery.

Troubleshooting Commands

The JUNOS software includes commands that are useful when troubleshooting a problem. This section provides a brief description of each command followed by sample output, and a discussion of the output in relation to the problem.

The following commands can be used when troubleshooting a fast reroute error in an MPLS-VPN network:

user@R2-PE1> **show configuration routing-instances vpn-a**
user@R2-PE1> **show configuration routing-options**
user@R2-PE1> **show bgp summary instance vpn-a**
user@R2-PE1> **show configuration protocols mpls**
user@R2-PE1> **show mpls lsp ingress**
user@R2-PE1> **show rsvp session ingress**
user@R2-PE1> **show rsvp session ingress detail**
user@R2-PE1> **show route table vpn-a 192.168.5.1 detail**

user@R2-PE1> **show route forwarding-table vpn vpn-a destination 192.168.5.1 extensive**

Sample Output The show configuration *statement-path* command is used to display a specific configuration hierarchy; in this case, to verify the correct configuration of a specific routing instance named vpn-a.

```
user@R2-PE1> show configuration routing-instances vpn-a
instance-type vrf;
interface so-0/0/0.0;
vrf-target {
    import target:65432:100;
    export target:65432:100;
}
protocols {
    bgp {
        group CE1 {
            type external;
            peer-as 65400;
            neighbor 10.0.12.1;
        }
    }
}
```

What It Means The sample output for the show configuration command shows the current running configuration of the specific routing instance named vpn-a configured on the ingress PE1 router. The vpn-a instance configuration has a VRF table that supports EBGP routing on the PE-CE link (so-0/0/0.0). This interface is the correct interface for the CE1-PE1 link in the network topology shown in Figure 14.

The VRF instance is linked to a VFR target community configured at the [edit policy-options] hierarchy level, allowing advertising of L3 VPN routes between PE routers. (See the PE1 configuration in "Router Configurations" on page 138 for the policy options configuration.) The import statement places, into the vpn-a.inet.0 table, all received L3 VPN MP-BGP routes tagged with the correct target community. The export statement advertises and tags all routes in the vpn-a.inet.o table with the listed target community to all MP-BGP peers.

The BGP protocols configuration within the routing instance applies the BGP import and export policies to the exchange of BGP routes on the PE-CE routing instance.

Sample Output The show bgp summary command is used to display summary information about BGP and its neighbors to determine if routes are received from peers in the autonomous system (AS). In this case, information for the specified instance vpn-a is displayed.

```
user@R2-PE1> show bgp summary instance vpn-a
Groups: 1 Peers: 1 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
vpn-a.inet.0          11          7          0          0          0          0
Peer               AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn
State|#Active/Received/Damped...
10.0.12.1       65400       2471       2473       0       0    20:35:20 Establ
  vpn-a.inet.0: 5/5/0
```

What It Means The sample output for the show bgp summary instance vpn-a command shows that the peering session between the CE1 and PE1 routers is established, indicating that the peers are exchanging update messages.

Sample Output The show configuration *statement-path* command is used to display a specific configuration hierarchy; in this case, the MPLS hierarchy.

```
user@R2-PE1> show configuration protocols mpls
label-switched-path r2-r4 {
    to 192.168.4.1;
    link-protection;
    primary direct;
}
path direct {
    10.0.24.2 strict;
}
interface all;
interface fxp0.0 {
    disable;
}
```

What It Means The sample output for the show configuration protocols mpls command shows the current running MPLS configuration on the ingress PE1 router. The configuration include the LSP r2-r4, link protection, and the strict primary path direct.

Sample Output The show mpls lsp command is used to display summarized information about the configured and active LSPs on a router; in this case, the command shows only the ingress LSPs on the ingress PE1 router.

```
user@R2-PE1> show mpls lsp ingress
Ingress LSP: 1 sessions
To              From            State Rt ActivePath       P     LSPname
192.168.4.1     192.168.2.1     Up     0 direct           *     r2-r4
Total 1 displayed, Up 1, Down 0
```

What It Means The sample output for the show mpls lsp ingress command shows that the ingress LSP r2-r4 is up and following the configured path direct.

Sample Output The show rsvp session command is used to display summarized information about active RSVP sessions on a router; in this case, the command shows summarized information about ingress RSVP sessions on the PE1 router

```
user@R2-PE1> show rsvp session ingress
Ingress RSVP: 2 sessions
To              From            State   Rt Style Labelin Labelout LSPname
192.168.4.1     192.168.2.1     Up       0  1 SE       -        3 r2-r4
192.168.4.1     192.168.2.1     Up       0  1 SE       -   100064
Bypass->10.0.24.2
Total 2 displayed, Up 2, Down 0
```

What It Means The sample output for the show rsvp session ingress command shows two RSVP sessions are up; the main LSP r2-r4 and a bypass path protecting the main LSP. Both RSVP sessions are in the Shared Explicit (SE) style, creating a shared reservation among for the two paths.

Sample Output The show rsvp session ingress detail command is used to display more detailed information about the two ingress RSVP sessions on the PE1 router.

```
user@R2-PE1> show rsvp session ingress detail
Ingress RSVP: 2 sessions

192.168.4.1
  From: 192.168.2.1, LSPstate: Up, ActiveRoute: 0
```

```
  LSPname: r2-r4, LSPpath: Primary
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: 3
  Resv style: 1 SE, Label in: -, Label out: 3
  Time left:    -, Since: Fri Mar  9 14:05:03 2007
  Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
  Port number: sender 1 receiver 63395 protocol 0
  Link protection desired
  Type: Link protected LSP
  PATH rcvfrom: localclient
  Adspec: sent MTU 1500
  Path MTU: received 1500
  PATH sentto: 10.0.24.2 (so-0/0/1.0) 2008 pkts
  RESV rcvfrom: 10.0.24.2 (so-0/0/1.0) 2006 pkts
  Explct route: 10.0.24.2
  Record route: <self> 10.0.24.2

192.168.4.1
  From: 192.168.2.1, LSPstate: Up, ActiveRoute: 0
  LSPname: Bypass->10.0.24.2
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: 100064
  Resv style: 1 SE, Label in: -, Label out: 100064
  Time left:    -, Since: Fri Mar  9 14:05:58 2007
  Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
  Port number: sender 1 receiver 63396 protocol 0
  Type: Bypass LSP
    Number of data route tunnel through: 1
    Number of RSVP session tunnel through: 0
  PATH rcvfrom: localclient
  Adspec: sent MTU 1500
  Path MTU: received 1500
  PATH sentto: 10.0.23.14 (fe-0/1/1.0) 2001 pkts
  RESV rcvfrom: 10.0.23.14 (fe-0/1/1.0) 1736 pkts
  Explct route: 10.0.23.14 10.0.34.14
  Record route: <self> 10.0.23.14 10.0.34.14
Total 2 displayed, Up 2, Down 0
```

What It Means The sample output for the show rsvp session ingress detail command shows the RSVP session for the ingress LSP and the bypass path, which appears as a separate RSVP ingress session for the protected interface 10.0.24.2. The bypass path is automatically generated. By default, the name appears as Bypass > *interface-address,* where the interface address is the next downstream router's interface (10.0.24.2). The explicit route 10.0.23.14 10.0.34.14 for the session shows R3 as the transit node and R4 as the egress node.

Sample Output The show route table *routing-table-name* command is used to display information about a particular routing table. In this case, the vpn-a.inet.0 routing table.

```
user@R2-PE1> show route table vpn-a 192.168.5.1 detail

vpn-a.inet.0: 9 destinations, 13 routes (9 active, 0 holddown, 0 hidden)
192.168.5.0/24 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 192.168.4.1:4
                Next-hop reference count: 11
                Source: 192.168.4.1
                Next hop: via so-0/0/1.0 weight 0x1, selected
                Label-switched-path r2-r4
                Label operation: Push 100160
                Next hop: 10.0.23.14 via fe-0/1/1.0 weight 0x8001
```

```
                    Label-switched-path r2-r4
                    Label operation: Push 100160, Push 100064(top)
                    Protocol next hop: 192.168.4.1
                    Push 100160
                    Indirect next hop: 8791000 262142
                    State: <Secondary Active Int Ext>
                    Local AS: 65432 Peer AS: 65432
                    Age: 1d 5:22:31         Metric2: 1
                    Task: BGP_65432.192.168.4.1+2056
                    Announcement bits (1): 0-KRT
                    AS path: 65400 I
                    Communities: target:65432:100
                    VPN Label: 100160
                    Localpref: 100
                    Router ID: 192.168.4.1
                    Primary Routing Table bgp.l3vpn.0
```

What It Means The sample output for the show route table vpn-a 192.168.5.1 detail command shows routes associated with the remote PE-CE location as indicated by the loopback address of the PE2 router 192.168.5.1. In this case, there are different next hops with unequal weights (0x1 and 0x8001) associated with the remote location. For correct traffic protection, those two routes must appear in the forwarding table.

Sample Output The show route forwarding-table command displays the route entries in the kernel's forwarding table. This is the version of the forwarding table in the Routing Engine. The Routing Engine copies this table to the Packet Forwarding Engine. In this case, the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information to the forwarding table for the specified destination.

```
user@R2-PE1> show route forwarding-table vpn vpn-a destination 192.168.5.1
extensive
Routing table: vpn-a.inet [Index 2]
Internet:

Destination:  192.168.5.0/24
  Route type: user
  Route reference: 0                    Route interface-index: 0
  Flags: sent to PFE, prefix load balance
  Next-hop type: indirect               Index: 262142   Reference: 2
  Next-hop type: Push 100160
  Next-hop interface: so-0/0/1.0
```

What It Means The sample output for the show route forwarding-table vpn vpn-a destination 192.168.5.1 extensive command shows only one next hop so-0/0/1.0 is installed in the forwarding table, indicating that the information in the forwarding table is not correct. We would expect to see the same paths installed in the forwarding table as appear in the routing table in the output for the show route table vpn-a 192.168.5.1 detail.

Solution

The solution is to enable load-balancing and ensure that multiple next-hop forwarding table entries appear in the forwarding table for each destination. The forwarding-table entries can be an incoming MPLS label or an IP destination.

A load-balancing policy applied to the forwarding-table is the same mechanism required for ECMP (equal-cost multipath) load-balancing to install multiple next-hops into the forwarding-table. The extra paths installed for local repair are not used for load-balancing, because the paths are differently weighted, as demonstrated in the sample output for the show routing table and the show route forwarding-table commands.

NOTE: The load-balancing policy must be applied to all provider (P) and provider-edge (PE) routers that are required to support local repair.

The following sample output shows an example load-balancing configuration and the commands used to verify that the required two next-hop entries appear in the forwarding table.

Sample Output

Use the following two show configuration *statement-path* commands to display a specific configuration hierarchy; in this case, policy-options and routing-options.

```
user@R2-PE1> show configuration policy-options
policy-statement lbpf {
    then {
        load-balance per-packet;
    }
}
[...Output truncated...]

user@R2-PE1> show configuration routing-options
static {
    [...Output truncated...]
    route 100.100.1.0/24 reject;
}
router-id 192.168.2.1;
route-distinguisher-id 192.168.2.1;
autonomous-system 65432;
forwarding-table {
    export lbpf;
}
```

What It Means

The sample output for the show configuration policy-options and show configuration routing-options commands shows the two parts required to configure a load balancing policy. The lbpf policy includes the load-balance per-packet statement. The policy is then applied at the [edit routing options forwarding-table] hierarchy level with the export lbpf statement. Enabling load balancing results in the export of routes from the routing table to the forwarding table, and a solution to the problem.

NOTE: The load-balance per-packet statement is named *per-packet* for historical reasons. When the Packet Forwarding Engine was an IP Processor-1 (before JUNOS 4.0), JUNOS supported only per-packet load balancing. When the IP Processor-II was introduced the behavior was changed to per-flow load balancing without changing the statement.

Sample Output Use the show route forwarding-table command to display the Routing Engine's forwarding table, including the network-layer prefixes and their next hops. This command is used to help verify that the routing protocol process has relayed the correction information to the forwarding table. In this case, the option vpn *vpn* is used to display routing table entries for the specified VPN vpn-a.

```
user@R2-PE1> show route forwarding-table vpn vpn-a destination 192.168.5.1
extensive

Routing table: vpn-a.inet [Index 2]
Internet:

Destination:  192.168.5.0/24
  Route type: user
  Route reference: 0                    Route interface-index: 0
  Flags: sent to PFE
  Next-hop type: indirect               Index: 262142   Reference: 2
  Next-hop type: unilist                Index: 262146   Reference: 1
  Next-hop type: Push 100160
  Next-hop interface: so-0/0/1.0        Weight: 0x1
  Nexthop: 10.0.23.14
  Next-hop type: Push 100160, Push 100064(top)
  Next-hop interface: fe-0/1/1.0        Weight: 0x8001
```

What It Means The sample output for the show route forwarding-table vpn vpn-a destination 192.168.5.1 extensive command shows the correct two routes were relayed from the routing table to the forwarding table.

Conclusion

In conclusion, a load balancing policy is required for link protection to work effectively. The principles are the same for the configuration of the fast reroute and the node-link protection statements.

Router Configurations

Purpose Output that shows the configurations of all routers in the network. The no-more option entered after the pipe (|) prevents the output from being paginated if the output is longer than the length of the terminal screen.

Sample Output The following sample output is for the customer edge (CE) 1 router:

```
user@R1-CE1> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/0 {
        unit 0 {
            family inet {
                address 10.0.12.1/30;
            }
            family iso;
            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 {
        /* corperate and alpha net */
        route 172.16.0.0/12 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        /* old lab nets */
        route 192.168.0.0/16 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        route 0.0.0.0/0 {
            discard;
            retain;
            no-readvertise;
        }
        route 172.16.0.0/24 reject;
        route 172.16.1.0/24 reject;
        route 172.16.2.0/24 reject;
        route 172.16.3.0/24 reject;
        route 192.168.1.0/24 reject;
    }
    router-id 192.168.1.1;
    autonomous-system 65400;
}
protocols {
    bgp {
        group PE1 {
            type external;
            export stat;
            peer-as 65432;
            neighbor 10.0.12.2;
        }
    }
    ospf {
        traffic-engineering;
        export stat;
        area 0.0.0.0 {
            interface so-0/0/0.0;
            interface lo0.0 {
                passive;
            }
        }
    }
}
policy-options {
    policy-statement stat {
        term 1 {
            from protocol static;
            then accept;
```

```
        }
        term 2 {
            then reject;
        }
    }
}
```

Sample Output The following sample output is for the provider edge (PE) 1 ingress router :

```
user@R2-PE1> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/0 {
        description to-r1;
        unit 0 {
            family inet {
                address 10.0.12.2/30;
            }
            family iso;
            family mpls;
        }
    }
    so-0/0/1 {
        description to-r4;
        unit 0 {
            family inet {
                address 10.0.24.1/30;
            }
            family iso;
            family mpls;
        }
    }
    fe-0/1/1 {
        description to-r3;
        unit 0 {
            family inet {
                address 10.0.23.13/30;
            }
            family iso;
            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 {
        route 172.16.0.0/12 {
            next-hop 192.168.71.254;
            retain;
```

```
            no-readvertise;
        }
        route 192.168.0.0/16 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        route 0.0.0.0/0 {
            discard;
            retain;
            no-readvertise;
        }
        route 100.100.1.0/24 reject;
    }
    router-id 192.168.2.1;
    route-distinguisher-id 192.168.2.1;
    autonomous-system 65432;
    forwarding-table {
        export lbpf;
    }
}
protocols {
    rsvp {
        interface fxp0.0 {
            disable;
        }
        interface all {
            link-protection;
        }
    }
    mpls {
        label-switched-path r2-r4 {
            to 192.168.4.1;
            link-protection;
            primary direct;
        }
        path via-r3 {
            10.0.23.14 strict;
            10.0.34.14 strict;
        }
        path direct {
            10.0.24.2 strict;
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        export send-statics;
        group ibgp {
            type internal;
            local-address 192.168.2.1;
            family inet {
                unicast;
            }
            family inet-vpn {
                unicast;
            }
            export next-hop-self;
            peer-as 65432;
            neighbor 192.168.4.1;
        }
```

```
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface lo0.0 {
                passive;
            }
            interface fe-0/1/1.0;
            interface so-0/0/1.0;
        }
    }
}
policy-options {
    policy-statement lbpf {
        then {
            load-balance per-packet;
        }
    }
    policy-statement next-hop-self {
        from route-type external;
        then {
            next-hop self;
        }
    }
    policy-statement send-statics {
        term statics {
            from {
                route-filter 100.100.1.0/24 exact;
            }
            then accept;
        }
    }
    policy-statement vpna-export {
        term 1 {
            from protocol static;
            then {
                community add vpna-target;
                community add vpna-origin;
                accept;
            }
        }
        term 2 {
            then reject;
        }
    }
    policy-statement vpna-import {
        term 1 {
            from {
                protocol bgp;
                community vpna-target;
            }
            then accept;
        }
        term 2 {
            then reject;
        }
    }
    community vpna-origin members origin:192.168.2.1:1;
    community vpna-target members target:65432:100;
}
routing-instances {
    vpn-a {
        instance-type vrf;
```

```
        interface so-0/0/0.0;
        vrf-target {
            import target:65432:100;
            export target:65432:100;
        }
        protocols {
            bgp {
                group CE1 {
                    type external;
                    peer-as 65400;
                    neighbor 10.0.12.1;
                }
            }
        }
    }
}
```

Sample Output The following sample output is for the provider (P) transit router:

```
user@R3-P> show configuration | no-more
[...Output truncated...]
interfaces {
    fe-1/3/0 {
        description to-r4;
        unit 0 {
            family inet {
                address 10.0.34.13/30;
            }
            family iso;
            family mpls;
        }
    }
    fe-1/3/1 {
        description to-r2;
        unit 0 {
            family inet {
                address 10.0.23.14/30;
            }
            family iso;
            family mpls;
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.145/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.1/32;
            }
            family iso {
                address 49.0004.1921.6800.3001.00;
            }
        }
    }
}
routing-options {
    static {
```

```
            /* corperate and alpha net */
            route 172.16.0.0/12 {
                next-hop 192.168.71.254;
                retain;
                no-readvertise;
            }
            /* old lab nets */
            route 192.168.0.0/16 {
                next-hop 192.168.71.254;
                retain;
                no-readvertise;
            }
            route 0.0.0.0/0 {
                discard;
                retain;
                no-readvertise;
            }
        }
        router-id 192.168.3.1;
        autonomous-system 65432;
    }
    protocols {
        rsvp {
            interface all {
                link-protection;
            }
            interface fxp0.0 {
                disable;
            }
        }
        mpls {
            icmp-tunneling;
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
        ospf {
            traffic-engineering;
            area 0.0.0.0 {
                interface lo0.0 {
                    passive;
                }
                interface fxp0.0 {
                    disable;
                }
                interface all;
            }
        }
    }
```

Sample Output The following sample output is for the provider edge (PE) 2 ingress router :

```
user@R4-PE2> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/1 {
        description to-R2;
        unit 0 {
            family inet {
                address 10.0.24.2/30;
            }
```

```
            family iso;
            family mpls;
        }
    }
    so-0/0/2 {
        description to-R5-CE2;
        unit 0 {
            family inet {
                address 10.0.45.1/30;
            }
            family iso;
            family mpls;
        }
    }
    fe-0/1/3 {
        description to-R3-P;
        unit 0 {
            family inet {
                address 10.0.34.14/30;
            }
            family iso;
            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 {
        route 172.16.0.0/12 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        route 192.168.0.0/16 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        route 0.0.0.0/0 {
            discard;
            retain;
            no-readvertise;
        }
        route 100.100.4.0/24 reject;
    }
    router-id 192.168.4.1;
    route-distinguisher-id 192.168.4.1;
    autonomous-system 65432;
    forwarding-table {
        export lbpf;
```

```
    }
}
protocols {
    rsvp {
        interface fxp0.0 {
            disable;
        }
        interface all {
            link-protection;
        }
    }
    mpls {
        label-switched-path r4-r2 {
            to 192.168.2.1;
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        export send-statics;
        group ibgp {
            type internal;
            local-address 192.168.4.1;
            family inet {
                unicast;
            }
            family inet-vpn {
                unicast;
            }
            export next-hop-self;
            peer-as 65432;
            neighbor 192.168.2.1;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface lo0.0 {
                passive;
            }
            interface fe-0/1/3.0;
            interface so-0/0/1.0;
        }
    }
}
policy-options {
    policy-statement lbpf {
        then {
            load-balance per-packet;
        }
    }
    policy-statement next-hop-self {
        from route-type external;
        then {
            next-hop self;
        }
    }
    policy-statement send-statics {
        term statics {
            from {
                route-filter 100.100.4.0/24 exact;
```

```
            }
            then accept;
        }
    }
    policy-statement vpnb-export {
        term 1 {
            from protocol static;
            then {
                community add vpnb-target;
                community add vpnb-origin;
                accept;
            }
        }
        term 2 {
            then reject;
        }
    }
    policy-statement vpnb-import {
        term 1 {
            from {
                protocol bgp;
                community vpnb-target;
            }
            then accept;
        }
        term 2 {
            then reject;
        }
    }
    community vpnb-origin members origin:192.168.5.1:1;
    community vpnb-target members target:65432:100;
}
routing-instances {
    vpn-b {
        instance-type vrf;
        interface so-0/0/2.0;
        vrf-target {
            import target:65432:100;
            export target:65432:100;
        }
        protocols {
            bgp {
                group CE2 {
                    type external;
                    peer-as 65400;
                    neighbor 10.0.45.2;
                }
            }
        }
    }
}
```

Sample Output The following sample output is for the customer edge (CE) 2 router:

```
user@R5-CE2> show configuration | no-more
[...Output truncated...]
interfaces {
    so-0/0/2 {
        unit 0 {
            family inet {
                address 10.0.45.2/30;
            }
```

```
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.70.147/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.5.1/32;
            }
            family iso {
                address 49.0004.1921.6800.5001.00;
            }
        }
    }
}
routing-options {
    graceful-restart;
    static {
        /* corperate and alpha net */
        route 172.16.0.0/12 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        /* old lab nets */
        route 192.168.0.0/16 {
            next-hop 192.168.71.254;
            retain;
            no-readvertise;
        }
        route 0.0.0.0/0 {
            discard;
            retain;
            no-readvertise;
        }
        route 172.16.0.0/24 reject;
        route 172.16.1.0/24 reject;
        route 172.16.2.0/24 reject;
        route 172.16.3.0/24 reject;
        route 192.168.5.0/24 reject;
    }
    router-id 192.168.5.1;
    autonomous-system 65400;
}
protocols {
    bgp {
        group PE2 {
            type external;
            export stat;
            peer-as 65432;
            neighbor 10.0.45.1;
        }
    }
    ospf {
        traffic-engineering;
        export stat;
        area 0.0.0.0 {
            interface so-0/0/2.0;
```

```
            interface lo0.0 {
                passive;
            }
        }
    }
}
policy-options {
    policy-statement stat {
        term 1 {
            from protocol static;
            then accept;
        }
        term 2 {
            then reject;
        }
    }
}
```