

Chapter 10

Configuring Layer 3 VPNs

To configure Layer 3 virtual private network (VPN) functionality, you must enable VPN support on the provider edge (PE) router. You must also configure any provider (P) routers that service the VPN, and you must configure the customer edge (CE) routers so that their routes are distributed into the VPN.

To configure Layer 3 VPNs, you include the following statements:

```
description text;  
instance-type vrf;  
interface interface-name;  
route-distinguisher ( as-number:number | ip-address:number );  
vrf-export [ policy-names ];  
vrf-import [ policy-names ];  
vrf-target ( community | export community-name | import community-name );  
vrf-table-label;  
protocols {  
  bgp {  
    group group-name {  
      peer-as as-number;  
      neighbor ip-address;  
    }  
    multihop tvl-value;  
  }  
  (ospf | ospf3) {  
    area area {  
      interface interface-name;  
    }  
    domain-id domain-id;  
    domain-vpn-tag number;  
    sham-link {  
      local address;  
    }  
    sham-link-remote address <metric number>;  
  }  
  pim {  
    vpn-group-address address;  
  }  
  rip {  
    rip-configuration;  
  }  
}
```

```

routing-options {
  autonomous-system autonomous-system {
    independent-domain;
    loops number;
  }
  forwarding-table {
    export [ policy-names ];
  }
  interface-routes {
    rib-group group-name;
  }
  martians {
    destination-prefix match-type <allow>;
  }
  maximum-paths {
    path-limit;
    log-interval interval;
    log-only;
    threshold percentage;
  }
  maximum-prefixes {
    prefix-limit;
    log-interval interval;
    log-only;
    threshold percentage;
  }
  multipath {
    vpn-unequal-cost;
  }
  options {
    syslog (level level | upto level);
  }
  rib routing-table-name {
    martians {
      destination-prefix match-type <allow>;
    }
    multipath {
      vpn-unequal-cost;
    }
    static {
      defaults {
        static-options;
      }
      route destination-prefix {
        next-hop [next-hops];
        static-options;
      }
    }
  }
}
router-id address ;

```

```

static {
  defaults {
    static-options;
  }
  route destination-prefix {
    policy [ policy-names ];
    static-options;
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

For Layer 3 VPNs, only some of the statements in the [edit routing-instances] hierarchy are valid. For the full hierarchy, see the *JUNOS Routing Protocols Configuration Guide*.

In addition to these statements, you must enable a signaling protocol, internal Border Gateway Protocol (IBGP) sessions between the PE routers, and an interior gateway protocol (IGP) on the PE and P routers.

By default, Layer 3 VPNs are disabled.

Many of the configuration procedures for Layer 3 VPNs are common to all types of VPNs. These procedures are described in detail in “Configuring VPNs” on page 11 and include the following:

- Enabling a Signaling Protocol on the PE Routers on page 12
- Configuring an IGP on the PE and P Routers on page 15
- Configuring an IBGP Session Between PE Routers on page 16
- Configuring a VPN Routing Instance on the PE Routers on page 17
- Configuring Graceful Restart on page 33

This chapter describes how to configure Layer 3 VPNs, discussing the following topics:

- Configuring VPN Routing Between the PE and CE Routers on page 148
- Configuring Layer 3 VPNs to Carry IBGP Traffic on page 163
- Filtering Traffic Based on the IP Header on page 164
- Configuring a VPN Tunnel for VRF Table Lookup on page 171
- Configuring a Logical Unit on the Loopback Interface on page 171
- Configuring Multicast over Layer 3 VPNs on page 173

- [Configuring Packet Forwarding for Layer 3 VPNs on page 174](#)
- [Configuring GRE Tunnels for Layer 3 VPNs on page 175](#)
- [Configuring an ES Tunnel Interface for Layer 3 VPNs on page 178](#)
- [Configuring IPsec Instead of MPLS Between PE Routers on page 181](#)
- [Configuring SCU and DCU for Layer 3 VPNs on page 185](#)
- [Protocol-Independent Load Balancing for Layer 3 VPNs on page 185](#)
- [Configuring Layer 3 VPN Policing on Interfaces on page 187](#)
- [Sending RADIUS Messages Through a Layer 3 VPN on page 188](#)

For configuration examples, see “Layer 3 VPN Configuration Examples” on page 207 and “Layer 3 VPN Internet Access Examples” on page 319.

Configuring VPN Routing Between the PE and CE Routers

For the PE router to distribute VPN-related routes to and from connected CE routers, you must configure routing within the VPN routing instance. You can configure a routing protocol—Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), or Routing Information Protocol (RIP)—or you can configure static routing. For the connection to each CE router, you can configure only one type of routing.

The following sections explain how to configure VPN routing between the PE and CE routers:

- [Configuring BGP Between the PE and CE Routers on page 149](#)
- [Configuring OSPF Between the PE and CE Routers on page 149](#)
- [Configuring RIP Between the PE and CE Routers on page 156](#)
- [Configuring Static Routes Between the PE and CE Routers on page 158](#)
- [Limiting the Paths and Prefixes Accepted from a CE Router on page 158](#)
- [Configuring IPv6 Between the PE and CE Routers on page 159](#)
- [Configuring EBGP or IBGP Multihop Between PE and CE Routers on page 163](#)

Configuring BGP Between the PE and CE Routers

To configure BGP as the routing protocol between the PE and the CE routers, include the `bgp` statement:

```
bgp {
  group group-name {
    peer-as as-number;
    neighbor ip-address;
  }
}
```

You can include the `bgp` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]



NOTE: Route reflectors and cluster IDs are not supported on a routing instance. Do not configure the `cluster-id` statement at the [edit routing-instances *routing-instance-name* protocols `bgp group group-name`] hierarchy level. Doing so causes the configuration to fail.

Configuring OSPF Between the PE and CE Routers

You can configure OSPF (version 2 or version 3) to distribute VPN-related routes between PE and CE routers.

The following sections describe how to configure OSPF as a routing protocol between the PE and the CE routers:

- Configuring OSPF Version 2 Between the PE and CE Routers on page 150
- Configuring OSPF Version 3 Between the PE and CE Routers on page 150
- Configuring OSPF Sham Links for Layer 3 VPNs on page 150
- Configuring an OSPF Domain ID on page 154

Configuring OSPF Version 2 Between the PE and CE Routers

To configure OSPF version 2 as the routing protocol between a PE and CE router, include the `ospf` statement:

```
ospf {
  area area {
    interface interface-name;
  }
}
```

You can include the `ospf` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

Configuring OSPF Version 3 Between the PE and CE Routers

To configure OSPF version 3 as the routing protocol between a PE and CE router, include the `ospf3` statement:

```
ospf3 {
  area area {
    interface interface-name;
  }
}
```

You can include the `ospf3` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

Configuring OSPF Sham Links for Layer 3 VPNs

When you configure OSPF between the PE and CE routers of a Layer 3 VPN, you can also configure OSPF sham links to compensate for issues related to OSPF intraarea links.

The following sections describe OSPF sham links and how to configure them:

- OSPF Sham Links Overview on page 151
- Configuring OSPF Sham Links on page 151
- OSPF Sham Links Example on page 152

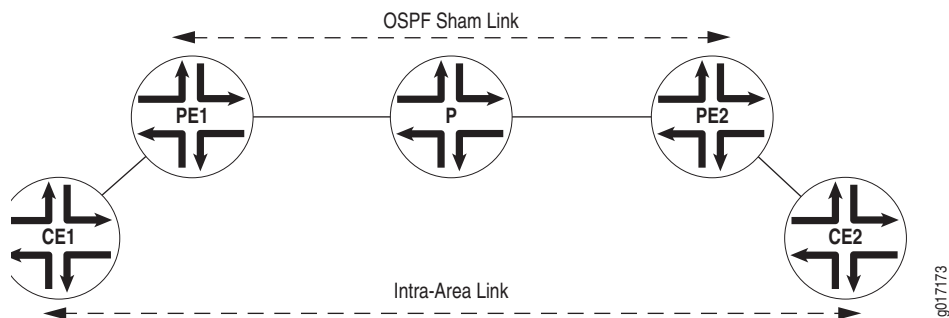
OSPF Sham Links Overview

Figure 19 provides an illustration of when you might configure an OSPF sham link. Router CE1 and Router CE2 are located in the same OSPF area. These CE routers are linked together by a Layer 3 VPN over Router PE1 and Router PE2. In addition, Router CE1 and Router CE2 are connected by an intraarea link used as a backup.

OSPF treats the link through the Layer 3 VPN as an interarea link. By default, OSPF prefers intraarea links to interarea links, so OSPF selects the backup intraarea link as the active path. This is not acceptable in configurations where the intraarea link is not the expected primary path for traffic between the CE routers.

An OSPF sham link is also an intraarea link, except that it is configured between the PE routers as shown in Figure 19. You can configure the metric for the sham link to ensure that the path over the Layer 3 VPN is preferred to a backup path over an intraarea link connecting the CE routers.

Figure 19: OSPF Sham Link



You should configure an OSPF sham link under the following circumstances:

- Two CE routers are linked together by a Layer 3 VPN.
- These CE routers are in the same OSPF area.
- An intraarea link is configured between the two CE routers.

If there is no intraarea link between the CE routers, you do not need to configure an OSPF sham link.

For more information on OSPF sham links, see the Internet draft [draft-ietf-l3vpn-ospf-2547-01.txt](#), *OSPF as the PE/CE Protocol in BGP/MPLS VPNs*.

Configuring OSPF Sham Links

The sham link is an unnumbered point-to-point intraarea link and is advertised by means of a type 1 link-state advertisement (LSA). Sham links are valid only for routing instances and OSPF version 2.

Each sham link is identified by a combination of the local and remote sham link end-point address and the OSPF area to which it belongs. Sham links must be configured manually. You configure the sham link between two PE routers, both of which are within the same VRF routing instance.

You need to specify the address for the local end point of the sham link. This address is used as the source for the sham link packets and is also used by the remote PE router as the sham link remote end-point.

The OSPF sham link's local address must be specified with a loopback address for the local VPN. The route to this address must be propagated by BGP. Specify the address for the local end point using the `local` option of the `sham-link` statement:

```
sham-link {
  local address;
}
```

You can include the `sham-link` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols ospf]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols ospf]

The OSPF sham link's remote address must be specified with a loopback address for the remote VPN. The route to this address must be propagated by BGP. To specify the address for the remote end point, include the `sham-link-remote` statement:

```
sham-link-remote address <metric number>;
```

You can include the `sham-link-remote` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols ospf area *area-id*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols ospf area *area-id*]

Optionally, you can include the `metric` option to set a metric value for the remote end point. The metric value specifies the cost of using the link. Routes with lower total path metrics are preferred over those with higher path metrics.

You can configure a value from 1 through 65,535. The default value is 1.

OSPF Sham Links Example

This example shows how to enable OSPF sham links on a PE router.

The following is the loopback interface configuration on the PE router. The address configured is for the local end point of the OSPF sham link:

```
[edit]
interfaces {
  lo0 {
    unit 1 {
      family inet {
        address 10.1.1.1/32;
      }
    }
  }
}
```

The following is the routing instance configuration on the PE router, including the configuration for the OSPF sham link. The `sham-link local` statement is configured with the address for the local loopback interface:

```
[edit]
routing-instances {
  example-sham-links {
    instance-type vrf;
    interface e1-1/0/2.0;
    interface lo0.1;
    route-distinguisher 3:4;
    vrf-import vpn-red-import;
    vrf-export vpn-red-export;
    protocols {
      ospf {
        sham-link local 1-1.1.1;
        area 0.0.0.0 {
          sham-link-remote 10.2.2.2 metric 1;
          interface e1-1/0/2.0 metric 1;
        }
      }
    }
  }
}
```

Configuring an OSPF Domain ID

For most OSPF configurations involving Layer 3 VPNs, you do not need to configure an OSPF domain ID. However, for a Layer 3 VPN connecting multiple OSPF domains, configuring OSPF domain IDs can help you control LSA translation (for Type 3 and Type 5 LSAs) between the OSPF domains and back-door paths. Each VPN routing and forwarding (VRF) table in a PE router associated with an OSPF instance is configured with the same OSPF domain ID. The default OSPF domain ID is the null value 0.0.0.0. As shown in Table 5, a route with a null domain ID is handled differently from a route without any domain ID at all.

Table 5: How a PE Router Redistributes and Advertises Routes

Route Received	Domain ID of the Route Received	Domain ID on the Receiving Router	Route Redistributed and Advertised As
Type 3 route	A.B.C.D	A.B.C.D	Type 3 LSA
Type 3 route	A.B.C.D	E.F.G.H	Type 5 LSA
Type 3 route	0.0.0.0	0.0.0.0	Type 3 LSA
Type 3 route	Null	0.0.0.0	Type 3 LSA
Type 3 route	Null	Null	Type 3 LSA
Type 3 route	0.0.0.0	Null	Type 3 LSA
Type 3 route	A.B.C.D	Null	Type 5 LSA
Type 3 route	Null	A.B.C.D	Type 5 LSA
Type 5 route	Not applicable	Not applicable	Type 5 LSA

You can configure an OSPF domain ID for both version 2 and version 3 of OSPF. The only difference in the configuration is that you include statements at the [edit routing-instances *routing-instance-name* protocols ospf] hierarchy level for OSPF version 2 and at the [edit routing-instances *routing-instance-name* protocols ospf3] hierarchy level for OSPF version 3. The configuration descriptions that follow present the OSPF version 2 statement only. However, the substatements are also valid for OSPF version 3.

To configure an OSPF domain ID, include the `domain-id` statement:

```
domain-id domain-ID;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols ospf]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols ospf]

You can set a VPN tag for the OSPF external routes generated by the PE router to prevent looping. By default, this tag is automatically calculated and needs no configuration. However, you can configure the domain VPN tag for Type 5 LSAs explicitly by including the `domain-vpn-tag number` statement:

```
domain-vpn-tag number;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols ospf]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols ospf]

The range is 1 through 4,294,967,295 ($2^{32} - 1$). If you set VPN tags manually, you must set the same value for all PE routers in the VPN.

For an example of this type of configuration, see “Configuring an OSPF Domain ID for a Layer 3 VPN” on page 278.

Hub-and-Spoke Layer 3 VPNs and OSPF Domain ID

The default behavior of an OSPF domain ID can cause the following problems for hub-and-spoke Layer 3 VPNs using OSPF between the PE and CE routers:

- PE routers set the down (DN) bit on all OSPF summary LSAs originating from area 0. PE routers are designated as area 0 by default because of the OSPF domain ID. When a PE router receives a summary LSA with the DN bit set, the LSA is not used in the OSPF calculation. This is done to prevent routing loops.

For a hub-and-spoke Layer 3 VPN, when the hub PE router generates an OSPF summary LSA, it also sets the DN bit before sending it to the hub CE router. When the hub CE router sends the LSA back to the PE router, the PE router does not use the LSA in the OSPF calculation because the DN bit is set. Routes aggregated within the CE router are not affected.

- PE routers generating external LSAs learned from BGP updates set the `vpn-route-tag` field to a value derived from the PE router’s AS number and an arbitrary tag. When a PE router receives an external LSA with a `vpn-route-tag` field that matches its own `vpn-route-tag` field, the LSA is not used in the OSPF calculation. This is done to prevent routing loops.

For a hub-and-spoke Layer 3 VPN, an external LSA originated by a hub PE router is sent to the hub CE router, which then sends it back to the same PE router. Because the `vpn-route-tag` field matches the PE router’s `vpn-route-tag` field, the LSA is not used in the OSPF calculation. Routes aggregated within the CE router are not affected.

For hub-and-spoke Layer 3 VPNs using OSPF between the PE and CE routers to work, you need to configure the following on the hub PE router:

- Configure the `disable` statement at the `[edit routing-instances routing-instance-name protocols ospf domain-id]` hierarchy level on the routing instance for the hub CE router. This removes area 0 from the PE router, allowing the PE router to forward LSAs without setting the DN bit. When an LSA comes back from the hub CE router, the PE router can install it because the DN bit is not set.
- Configure 0 for the `vpn-route-tag` statement at the `[edit routing-instances routing-instance-name protocols ospf]` hierarchy level on the routing instance for the spoke CE router. This removes any VPN route tags that are set on the external LSAs, preventing a VPN route tag match and allowing the PE router to install the LSA.

Configuring RIP Between the PE and CE Routers

For a Layer 3 VPN, you can configure RIP on the PE router to learn the routes of the CE router or to propagate the routes of the PE router to the CE router. RIP routes learned from neighbors configured at any `[edit routing-instances]` hierarchy level are added to the routing instance's `inet` table (`instance_name.inet.0`).

To configure RIP as the routing protocol between the PE and the CE router, include the `rip` statement:

```
rip {
  group group-name {
    export policy-names;
    neighbor interface-name;
  }
}
```

You can include the `rip` statement at the following hierarchy levels:

- `[edit routing-instances routing-instance-name protocols]`
- `[edit logical-routers logical-router-name routing-instances routing-instance-name protocols]`

By default, RIP does not advertise the routes it receives. To advertise routes from a PE router to a CE router, you need to configure an `export` policy on the PE router for RIP. For information on how to define an export policy, see the *JUNOS Policy Framework Configuration Guide*.

To specify an export policy for RIP, include the `export` statement:

```
export [ policy-names ];
```

You can include the `export` statement for RIP at the following hierarchy levels:

- `[edit routing-instances routing-instance-name protocols rip group group-name]`
- `[edit logical-routers logical-router-name routing-instances routing-instance-name protocols rip group group-name]`

To install routes learned from a RIP routing instance into multiple routing tables, include the `rib-group` and `group` statements:

```
rib-group inet group-name;  
group group-name {  
    neighbor interface-name;  
}
```

You can include these statements at the following hierarchy levels:

- [edit protocols]
- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

To configure a routing table group, include the `rib-groups` statement:

```
rib-groups group-name;
```

You can include the `rib-group` statement at the following hierarchy levels:

- [edit routing-options]
- [edit logical-routers *logical-router-name* routing-options]

To add a routing table to a routing table group, include the `import-rib` statement. The first routing table name specified under the `import-rib` statement must be the name of the routing table you are configuring. For more information about how to configure routing tables and routing table groups, see the *JUNOS Routing Protocols Configuration Guide*.

```
import-rib [ group-names ]
```

You can include the `import-rib` statement at the following hierarchy levels:

- [edit routing-options rib-groups *group-name*]
- [edit logical-routers *logical-router-name* routing-options rib-groups *group-name*]

Configuring Static Routes Between the PE and CE Routers

To configure a static route between the PE and the CE routers, include the `static` statement:

```
static {
  route destination-prefix {
    next-hop [ next-hops ];
    static-options;
  }
}
```

You can include the `static` statement at the following hierarchy levels:

- [edit routing-options]
- [edit logical-routers *logical-router-name* routing-options]

For more information about configuring routing protocols and static routes, see the *JUNOS Routing Protocols Configuration Guide*.

Limiting the Paths and Prefixes Accepted from a CE Router

You can configure a maximum limit on the number of prefixes and paths that can be installed into the routing tables. Using prefix and path limits, you can curtail the number of prefixes and paths received from a CE router in a VPN. Prefix and path limits apply only to dynamic routing protocols, and are not applicable to static or interface routes.

To limit the number of paths accepted by a PE router from a CE router, include the `maximum-paths` statement:

```
maximum-paths path-limit <log-interval interval | log-only | threshold percentage>;
```

For a list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

To limit the number of prefixes accepted by a PE router from a CE router, include the `maximum-prefixes` statement:

```
maximum-prefixes prefix-limit <log-interval interval | log-only | threshold percentage>;
```

For a list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

A mandatory path or prefix limit, in addition to triggering a warning message, rejects any additional paths or prefixes once the limit is reached.



NOTE: Setting a path or prefix limit might result in unpredictable dynamic routing protocol behavior.

You can also configure the following options for both the `maximum-paths` and `maximum-prefixes` statements:

- `log-interval`—Specify the interval at which log messages are sent.
- `log-only`—Generate warning messages only. No limit is placed on the number of paths or prefixes stored in the routing tables.
- `threshold`—Generate warning messages after the specified percentage of the maximum paths or prefixes has been reached.

Configuring IPv6 Between the PE and CE Routers

You can configure IP version 6 (IPv6) between the PE and CE routers of a Layer 3 VPN. The PE router must have the PE router to PE router BGP session configured with the `family inet6-vpn` statement. The CE router must be capable of receiving IPv6 traffic. You can configure BGP or static routes between the PE and CE routers.

The following sections explain how to configure IPv6 VPNs between the PE routers:

- Configuring IPv6 on the PE Router on page 159
- Configuring the Connection Between the PE and CE Routers on page 160
- Configuring IPv6 on the Interfaces on page 162

Configuring IPv6 on the PE Router

To configure IPv6 between the PE and CE routers, include the `family inet6-vpn` statements on the PE router:

```
family inet6-vpn {
  (any | multicast | unicast) {
    aggregate-label community community-name;
    prefix-limit maximum prefix-limit;
    rib-group rib-group-name;
  }
}
```

For a list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

You also must include the `ipv6-tunneling` statement:

```
ipv6-tunneling;
```

You can include the `ipv6-tunneling` statement at the following hierarchy levels:

- [edit protocols mpls]
- [edit logical-routers *logical-router-name* protocols mpls]

Configuring the Connection Between the PE and CE Routers

To support IPv6 routes, you must configure BGP, OSPF version 3, or static routes for the connection between the PE and CE routers in the Layer 3 VPN. You can configure BGP to handle just IPv6 routes or both IP version 4 (IPv4) and IPv6 routes.

For more information about IPv6, see the *JUNOS Routing Protocols Configuration Guide*.

The following sections explain how to configure BGP and static routes:

- Configuring BGP on the PE Router to Handle IPv6 Routes on page 160
- Configuring BGP on the PE Router for IPv4 and IPv6 Routes on page 161
- Configuring OSPF Version 3 on the PE Router on page 161
- Configuring Static Routes on the PE Router on page 162

Configuring BGP on the PE Router to Handle IPv6 Routes

To configure BGP in the Layer 3 VPN routing instance to handle IPv6 routes, include the `bgp` statement:

```

bgp {
  group group-name {
    local-address IPv6-address;
    family inet6 {
      unicast;
    }
    peer-as as-number;
    neighbor IPv6-address;
  }
}

```

You can include the `bgp` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

Configuring BGP on the PE Router for IPv4 and IPv6 Routes

To configure BGP in the Layer 3 VPN routing instance to handle both IPv4 and IPv6 routes, include the `bgp` statement:

```
bgp {
  group group-name {
    local-address IPv4-address;
    family inet {
      unicast;
    }
    family inet6 {
      unicast;
    }
    peer-as as-number;
    neighbor address;
  }
}
```

You can include the `bgp` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

Configuring OSPF Version 3 on the PE Router

To configure OSPF version 3 in the Layer 3 VPN routing instance to handle IPv6 routes, include the `ospf3` statement:

```
ospf3 {
  area area-id {
    interface interface-name;
  }
}
```

You can include the `ospf3` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols]

For complete configuration guidelines for this statement, see the *JUNOS Routing Protocols Configuration Guide*.

Configuring Static Routes on the PE Router

To configure a static route to the CE router in the Layer 3 VPN routing instance, include the `routing-options` statement:

```
routing-options {
  rib routing-table.inet6.0 {
    static {
      defaults {
        static-options;
      }
    }
  }
}
```

You can include the `routing-options` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

Configuring IPv6 on the Interfaces

You need to configure IPv6 on the PE router interfaces to the CE routers and on the CE router interfaces to the PE routers.

To configure the interface to handle IPv6 routes, include the `family inet6` statement:

```
family inet6 {
  address ipv6-address;
}
```

You can include the `family inet6` statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *unit-number*]
- [edit logical-routers *logical-router-name* interfaces *interface-name* unit *unit-number*]

If you have configured the Layer 3 VPN to handle both IPv4 and IPv6 routes, configure the interface to handle both IPv4 and IPv6 routes by including the `unit` statement:

```
unit unit-number {
  family inet {
    address ipv4-address;
  }
  family inet6 {
    address ipv6-address;
  }
}
```

You can include the `unit` statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]

Configuring EBGP or IBGP Multihop Between PE and CE Routers

You can configure an external BGP (EBGP) or IBGP multihop session between the PE and CE routers of a Layer 3 VPN. This allows you to have one or more routers between the PE and CE routers. Using IBGP between PE and CE routers does not require the configuration of any additional statements. However, using EBGP between the PE and CE routers requires the configuration of the `multihop` statement.

To configure an external BGP multihop session for the connection between the PE and CE routers, include the `multihop` statement on the PE router. To help prevent routing loops, you have to configure a time-to-live (TTL) value for the multihop session:

```
multihop ttl-value;
```

For the list of hierarchy levels at which you can configure this statement, see the summary section for this statement.

Configuring Layer 3 VPNs to Carry IBGP Traffic

When you configure BGP as the routing protocol between a PE router and a CE router in a Layer 3 VPN, you typically configure external peering sessions between the Layer 3 VPN service provider and the customer network ASs.

If the customer network has several sites advertising routes through an external BGP session to the service provider network and if the same AS is used by all the customer sites, the CE routers reject routes from the other CE routers. They detect a loop in the BGP AS path attribute.

To prevent the CE routers from rejecting each other's routes, you could configure the following:

- PE routers advertising routes received from remote PE routers can remap the customer network AS number to its own AS number.
- AS path loops can be configured.
- The customer network can be configured with different AS numbers at each site.

These types of configurations can work when there are no BGP routing exchanges between the customer network and other networks. However, they do have limitations for customer networks that use BGP internally for purposes other than carrying traffic between the CE routers and the PE routers. When those routes are advertised outside the customer network, the service provider ASs are present in the AS path.

To improve the transparency of Layer 3 VPN services for customer networks, you can configure the routing instance for the Layer 3 VPN to isolate the customer's network attributes from the service provider's network attributes.

When you include the `independent-domain` statement in the Layer 3 VPN routing instance configuration, BGP attributes received from the customer network (from the CE router) are stored in a BGP attribute (ATTRSET) that functions like a stack. When that route is advertised from the remote PE router to the remote CE router, the original BGP attributes are restored. This is the default behavior for BGP routes that are advertised to Layer 3 VPNs located in different domains.

This functionality is described in the Internet draft `draft-marques-ppvnp-ibgp-version.txt`, *RFC 2547bis Networks Using Internal BGP as PE-CE Protocol*.

To allow a Layer 3 VPN to transport IBGP traffic, include the `independent-domain` statement:

```
independent-domain;
```

You can include the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* routing-options autonomous-system *number*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name* routing-options autonomous-system *number*]



NOTE: All PE routers participating in a Layer 3 VPN configured with the `independent-domain` statement must be running JUNOS 6.3 or later.

Filtering Traffic Based on the IP Header

The `vrf-table-label` statement makes it possible to map the inner label to a specific VRF routing table; such mapping allows the examination of the encapsulated IP header at an egress VPN router. You might want to enable this functionality so that you can do either of the following:

- Forward traffic on a PE-router-to-CE-device interface, in a shared medium, where the CE device is a Layer 2 switch without IP capabilities (for example, a metro Ethernet switch).

The first lookup is done on the VPN label to determine which VRF table to refer to, and the second lookup is done on the IP header to determine how to forward packets to the correct end hosts on the shared medium.

- Perform egress filtering at the egress PE router.

The first lookup on the VPN label is done to determine which VRF routing table to refer to, and the second lookup is done on the IP header to determine how to filter and forward packets. You can enable this functionality by configuring output filters on the VRF interfaces.

When you use the `vrf-table-label` statement to configure a VRF routing table, a label-switched interface (LSI) logical interface label is created and mapped to the VRF routing table.

Any routes configured in a VRF routing table with the `vrf-table-label` statement are advertised with the LSI logical interface label allocated for the VRF routing table. When packets for this VPN arrive on a core-facing interface, they are treated as if the enclosed IP packet arrived on the LSI interface and are then forwarded and filtered based on the correct table.

The following sections describe how filter traffic based on the IP header:

- Configuring Traffic Filtering Based on the IP Header on page 165
- Applying MPLS EXP Classifiers to Routing Instances on page 170

Configuring Traffic Filtering Based on the IP Header

To filter traffic based on the IP header, include the `vrf-table-label` statement:

```
vrf-table-label;
```

You can include the `vrf-table-label` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

You can configure the `vrf-table-label` statement for both IPv4 and IPv6 Layer 3 VPNs. If you configure the `vrf-table-label` statement for a dual-stack VRF routing table (where both IPv4 and IPv6 routes are supported), the `vrf-table-label` statement applies to both the IPv4 and IPv6 routes and the same label is advertised for both sets of routes.



NOTE: When you configure the `vrf-table-label` statement, the TTL in the MPLS header is removed from packets sent from the PE router to the CE router.

Egress Filtering Options

You can enable egress filtering (which allows egress Layer 3 VPN PE routers to perform lookups on the VPN label and IP header at the same time) by including the `vrf-table-label` statement at the [edit routing-instances *instance-name*] hierarchy level. However, there are many limitations on when you can configure the `vrf-table-label` statement. For more information, see “Support for ATM and Frame Relay Interfaces” on page 167 and “Other Limitations” on page 168. There is no restriction on CE-router-to-PE-router interfaces.

You can also enable egress filtering by configuring a VPN tunnel (VT) interface on routing platforms equipped with a Tunnel Services Physical Interface Card (PIC). When you enable egress filtering this way, there is no restriction on the type of core-facing interface used. There is also no restriction on the type of CE-router-to-PE-router interface used.

Support for Ethernet, SONET/SDH, and T1/T3/E3 Interfaces

Support for the `vrf-table-label` statement over Ethernet, SONET/SDH, and DS3/T3 interfaces is available on the Juniper Networks routing platforms summarized in Table 6.

Table 6: Support for Ethernet and SONET/SDH Interfaces

Interfaces	J-series	M-series Without an Enhanced FPC	M-series with an Enhanced FPC	M320	T-series
Ethernet	Yes	Yes	Yes	Yes	Yes
SONET/SDH	N/A	Yes	Yes	Yes	Yes
T1/T3/E3	Yes	Yes	Yes	Yes	Yes

Only the following Ethernet PICs support the `vrf-table-label` statement on M-series routers without enhanced FPCs:

- 1-port Gigabit Ethernet
- 2-port Gigabit Ethernet
- 4-port Fast Ethernet

Support for Aggregated and VLAN Interfaces

Support for the `vrf-table-label` statement over aggregated and VLAN interfaces is available on the Juniper Networks routing platforms summarized in Table 7.

Table 7: Support for Aggregated and VLAN Interfaces

Interfaces	J-series	M-series Without an Enhanced FPC	M-series with an Enhanced FPC	M320	T-series
Aggregated	N/A	No	Yes	Yes	Yes
VLAN	Yes	No	Yes	Yes	Yes

Support for ATM and Frame Relay Interfaces

Support for the `vrf-table-label` statement over Asynchronous Transfer Mode (ATM) and Frame Relay interfaces is available on the Juniper Networks routing platforms summarized in Table 8.

Table 8: Support for ATM and Frame Relay Interfaces

Interfaces	J-series	M-series without an Enhanced FPC	M-series with an Enhanced FPC	M320	T-series
ATM1	N/A	No	No	No	No
ATM2 intelligent queuing (IQ)	N/A	No	Yes	Yes	Yes
Frame Relay	Yes	No	Yes	Yes	Yes
Channelized	N/A	No	No	No	No

When you configure the `vrf-table-label` statement, be aware of the following limitations with ATM or Frame Relay interfaces:

- The `vrf-table-label` statement is supported on ATM interfaces, but with the following limitations:
 - ATM interfaces can be configured on a T-series routing platform, on an M320, or on an M-series router fitted with an enhanced FPC.
 - The interface can only be a PE router interface receiving traffic from a P router.
 - The router must have an ATM2 IQ PIC.
- The `vrf-table-label` statement is also supported with Frame Relay encapsulated interfaces, but with the following limitations:
 - Frame Relay interfaces can be configured on a T-series routing platform, on an M320, or on an M-series router fitted with an enhanced FPC.
 - The interface can only be a PE router interface receiving traffic from a P router.

Support for Multilink PPP and Multilink Frame Relay Interfaces

Support for the `vrf-table-label` statement over Multilink Point-to-Point Protocol (MLPPP) and Multilink Frame Relay (MLFR) interfaces is available on the Juniper Networks routing platforms summarized in Table 9.

Table 9: Support for Multilink PPP and Multilink Frame Relay Interfaces

Interfaces	J-series	M-series Without an Enhanced FPC	M-series with an Enhanced FPC	M320	T-series
MLPPP	Yes	No	Yes	No	No
End-to-End MLFR (FRF.15)	Yes	No	Yes	No	No
UNI/NNI MLFR (FRF.16)	Yes	No	No	No	No

M-series routing platforms require an AS PIC to support the `vrf-table-label` statement over MLPPP and MLFR interfaces.

Support for Packets with Null Top Labels

You can configure the `vrf-table-label` statement on core-facing interfaces receiving MPLS packets with a null top label, which might be transmitted by some vendors' equipment. These packets can be received only on M320 and T-series routing platforms using one of the following interfaces:

- 1-port Gigabit Ethernet with SFP
- 1-port SONET STM4
- 1-port SONET STM16

Other Limitations

When you configure the `vrf-table-label` statement, be aware of the following other limitations:

- The time-to-live (TTL) value in the MPLS header is not copied back to the IP header of packets sent from the PE router to the CE router.
- You cannot configure a virtual loopback tunnel interface and the `vrf-table-label` statement on the same routing instance. Doing so causes the commit to fail.
- Do not use the `vrf-table-label` statement for source class usage/destination class usage (SCU/DCU) configurations. For information on SCU/DCU configuration, see the *JUNOS Network Interfaces Configuration Guide*.
- You can configure the `vrf-table-label` statement on Multilink Frame Relay (MLFR FRF.16) encapsulated PE-router-to-P-router interfaces, but only on J-series routing platforms.
- When you configure the `vrf-table-label` statement, MPLS packets with label-switched interface (LSI) labels that arrive on core-facing ATM or Frame Relay interfaces, or on aggregated Ethernet interfaces configured with VLANs or Ethernet interfaces configured with VLANs, are not counted at the logical interface level.

- You cannot configure the `vrf-table-label` statement within a VRF routing instance if the PE-router-to-P-router interface is any of the following:
 - Aggregated SONET/SDH interfaces
 - All channelized interfaces
 - All tunnel interfaces (for example, generic routing encapsulation [GRE] and IP Security [IPSec])
 - Circuit cross-connect (CCC) and translational cross-connect (TCC) encapsulated interfaces
 - Logical tunnel interfaces
 - Virtual private LAN service (VPLS) encapsulated interfaces



NOTE: All CE-router-to-PE-router and PE-router-to-CE-router interfaces are supported.

- You cannot configure the `vrf-table-label` statement within a VRF routing instance if the PE-router-to-P-router PIC is one of the following:
 - 10-port E1 PIC
 - 8-port Fast Ethernet PIC
 - 12-port Fast Ethernet PIC
 - 48-port Fast Ethernet PIC
 - All ATM PICs, except the ATM2 IQ PIC

Applying MPLS EXP Classifiers to Routing Instances

When you configure the `vrf-table-label` statement, and you do not explicitly apply a classifier configuration to the routing instance, the default MPLS EXP classifier is applied to the routing instance.

For PICs that are installed on Enhanced FPCs, you can override the default MPLS EXP classifier and apply a custom classifier to the routing instance. Detailed instructions for this procedure are provided in the *JUNOS Network Interfaces Configuration Guide*. The following instructions summarize how to apply a custom classifier to a routing instance:

1. Filter traffic based on the IP header by including the `vrf-table-label` statement at the `[edit routing-instances routing-instance-name]` hierarchy level:

```
vrf-table-label;
```

2. Configure a custom MPLS EXP classifier by including the following statements in the configuration. See the *JUNOS Network Interfaces Configuration Guide* for information on how to do this.
3. Configure the routing instance for CoS by including the `routing-instances` statement at the `[edit class-of-service]` hierarchy level:

```
routing-instances routing-instance-name {
  classifiers {
    exp (classifier-name | default);
  }
}
```

4. Configure the routing instance to use the custom MPLS EXP classifier by including the `classifiers` statement at the `[edit class-of-service routing-instances routing-instance-name]` hierarchy level:

```
classifiers {
  exp classifier-name;
}
```

To display the MPLS EXP classifiers associated with all routing instances, issue the `show class-of-service routing-instances` command.



NOTE: The following caveats apply to custom MPLS EXP classifiers for routing instances:

- An Enhanced FPC is required.
 - Logical routers are not supported.
-

Configuring a VPN Tunnel for VRF Table Lookup

You can configure a VPN tunnel to facilitate VRF table lookup based on MPLS labels. You might want to enable this functionality to forward traffic on a PE-router-to-CE-device interface in a shared medium, where the CE device is a Layer 2 switch without IP capabilities (for example, a metro Ethernet switch), or to perform egress filtering at the egress PE router.

For more information on VPN tunnels and VT interfaces, see the *JUNOS Services Interfaces Configuration Guide*.

Configuring a Logical Unit on the Loopback Interface

For Layer 3 VPNs (VRF routing instances), you can configure a logical unit on the loopback interface into each VRF routing instance that you have configured on the router. Associating a VRF routing instance with a logical unit on the loopback interface allows you to easily identify the VRF routing instance.

Doing this is useful for troubleshooting:

- It allows you to ping a remote CE router from a local PE router in a Layer 3 VPN. For more information, see “Pinging the Remote CE Router from the Local PE Router” on page 203.
- It ensures that a path maximum transmission unit (MTU) check on traffic originating on a VRF or virtual-router routing instance functions properly. For more information, see “Configuring a Path MTU Check for VPNs” on page 37.

You can also configure a firewall filter for the logical unit on the loopback interface; this configuration allows you to filter traffic for the VRF routing instance associated with it.

The following describes how firewall filters affect the VRF routing instance depending on whether they are configured on the default loopback interface, the VRF routing instance, or some combination of the two. The “default loopback interface” refers to `lo0.0` (associated with the default routing table), and the “VRF loopback interface” refers to `lo0.n`, which is configured in the VRF routing instance.

- If you configure Filter A on the default loopback interface and Filter B on the VRF loopback interface, the VRF routing instance uses Filter B.
- If you configure Filter A on the default loopback interface but do not configure a filter on the VRF loopback interface, the VRF routing instance does not use a filter.
- If you configure Filter A on the default loopback interface but do not even configure a VRF loopback interface, the VRF routing instance uses Filter A.

To configure a logical unit on the loopback interface, include the `unit` statement:

```
unit number {
  family inet {
    address address;
  }
}
```

You can include the `unit` statement at the following hierarchy levels:

- [edit interfaces lo0]
- [edit logical-routers *logical-router-name* interfaces lo0]

To associate a firewall filter with the logical unit on the loopback interface, include the `filter` statement:

```
filter {
  input filter-name;
}
```

You can include the `filter` statement at the following hierarchy levels:

- [edit interfaces lo0 unit *unit-number* family inet]
- [edit logical-routers *logical-router-name* interfaces lo0 unit *unit-number* family inet]

To include the `lo0.n` interface (where *n* specifies the logical unit) in the configuration for the VRF routing instance, include the following statement:

```
interface lo0.n;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

For more information on how to configure firewall filters, see the *JUNOS Policy Framework Configuration Guide*.

Configuring Multicast over Layer 3 VPNs

You can configure a Layer 3 VPN to support multicast traffic using the Protocol Independent Multicast (PIM) routing protocol. To support multicast, you need to configure PIM on routers within the VPN and within the service provider's network.

Each PE router configured to run multicast over Layer 3 VPNs must have a Tunnel Services PIC. A Tunnel Services PIC is also required on the P routers that act as rendezvous points (RPs). Tunnel Services PICs are also needed on all the CE routers acting as designated routers (first-hop/last-hop routers) or as RPs, just as they are in non-VPN PIM environments.

Configure the master PIM instance at the `[edit protocols pim]` hierarchy level on the CE and PE routers. This master PIM instance configuration on the PE router should match the configuration on the service providers core routers.

You also need to configure a PIM instance for the Layer 3 VPN at the `[edit routing-instances routing-instance-name protocols pim]` hierarchy level on the PE router. This creates a PIM instance for the indicated routing instance. The configuration of the PIM instance on the PE router should match the PIM instance configured on the CE router the PE router is connected to.

For information about how to configure PIM, see the *JUNOS Multicast Protocols Configuration Guide*.

You use the `vpn-group-address` statement to configure the group address designated for the VPN in the service provider's network. This address should be unique for each VPN and configured on the VRF routing instance of all PE routers connecting to the same VPN. It ensures that multicast traffic is transmitted only to the specified VPN.

Include the `vpn-group-address` statement:

```
vpn-group-address address;
```

You can include the `vpn-group-address` statement at the following hierarchy levels:

- `[edit routing-instances routing-instance-name protocols pim]`
- `[edit logical-routers logical-router-name routing-instances routing-instance-name protocols pim]`

The rest of the Layer 3 VPN configuration for multicast is conventional and is described in other sections of this manual. Most of the specific configuration tasks needed to activate multicast in a VPN environment involve PIM. For more information about how to configure PIM and multicast in JUNOS, including an example of how to configure multicast over Layer 3 VPNs, see the *JUNOS Multicast Protocols Configuration Guide*.

Configuring Packet Forwarding for Layer 3 VPNs

You can configure the router to support packet forwarding for IPv4 traffic in Layer 2 and Layer 3 VPNs. Packet forwarding is handled in one of the following ways, depending on the type of helper service configured:

- BOOTP service—Clients send Bootstrap Protocol (BOOTP) requests through the router configured with BOOTP service to a server in the specified routing instance. The server recognizes the client address and sends a response back to the router configured with BOOTP service. This router forwards the reply to the correct client address in the specified routing instance.
- Other services—Clients send requests through the router configured with the service to a server in the specified routing instance. The server recognizes the client address and sends a response to the correct client address in the specified routing instance.

To enable packet forwarding for VPNs, include the **helpers** statement:

```
helpers {
  service {
    description description-of-service;
    server {
      address address {
        routing-instance routing-instance-names;
      }
    }
    interface interface-name {
      description description-of-interface;
      no-listen;
      server {
        address address {
          routing-instance routing-instance-names;
        }
      }
    }
  }
}
```

You can include the **helpers** statement at the following hierarchy levels:

- [edit forwarding-options]
- [edit logical-routers *logical-router-name* forwarding-options]
- [edit routing-instances *routing-instance-name* forwarding-options]



NOTE: You can enable packet forwarding for multiple VPNs. However, the client and server must be within the same VPN. Any Juniper Networks routing platforms with packet forwarding enabled along the path between the client and server must also reside within the same VPN.

The address and routing instance together constitute a unique server. This has implications for routers configured with BOOTP service, which can accept multiple servers.

For example, a BOOTP service can be configured as follows:

```
[edit forwarding-options helpers bootp]
  server address 10.2.3.4 routing-instance [instance-A instance-B];
```

Even though the addresses are identical, the routing instances are different. A packet coming in for BOOTP service on **instance-A** is forwarded to 10.2.3.4 in the **instance-A** routing instance, while a packet coming in on **instance-B** is forwarded in the **instance-B** routing instance. Other services can only accept a single server, so this configuration does not apply in those cases.

For more information about the statements configured at the [edit forwarding-options] hierarchy level, see the *JUNOS Policy Framework Configuration Guide*.

Configuring GRE Tunnels for Layer 3 VPNs

JUNOS software allows you to configure a generic routing encapsulation (GRE) tunnel between the PE and CE routers for a Layer 3 VPN. The GRE tunnel can have one or more hops.

For more information about how to configure tunnel interfaces, see the *JUNOS Services Interfaces Configuration Guide*.

You can configure the GRE tunnels manually or configure the JUNOS software to instantiate GRE tunnels dynamically.

The following sections describe how to configure GRE tunnels manually and dynamically:

- Configuring GRE Tunnels Manually Between PE and CE Routers on page 175
- Configuring GRE Tunnels Dynamically on page 177

Configuring GRE Tunnels Manually Between PE and CE Routers

The following sections explain how to configure a GRE tunnel between the PE and CE routers for a Layer 3 VPN:

- Configuring the GRE Tunnel Interface on the PE Router on page 176
- Configuring the GRE Tunnel Interface on the CE Router on page 177

Configuring the GRE Tunnel Interface on the PE Router

You configure the GRE tunnel as a logical interface on the PE router. To configure the GRE tunnel interface, include the unit statement:

```

unit logical-unit-number {
  tunnel {
    source source-address;
    destination destination-address;
    routing-instance {
      destination routing-instance-name;
    }
  }
  family inet {
    address address;
  }
}

```

You can include the unit statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]

As part of the GRE tunnel interface configuration, you need to include the following statements:

- **source** *source-address*—Specify the source or origin of the GRE tunnel.
- **destination** *destination-address*—Specify the destination or end point of the GRE tunnel.

By default, the tunnel destination address is assumed to be in the default Internet routing table, *inet.0*. If the tunnel destination address is not in *inet.0*, you need to specify which routing table to search for the tunnel destination address by configuring the **routing-instance** statement. This is the case if the tunnel encapsulating interface is also configured under the routing instance.

- **destination** *routing-instance-name*—Specify the name of the routing instance when configuring the GRE tunnel interface on the PE router.

To complete the GRE tunnel interface configuration, include the **interface** statement for the GRE interface under the appropriate routing instance:

```

interface interface-name;

```

You can include the **interface** statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

Configuring the GRE Tunnel Interface on the CE Router

To configure the GRE tunnel interface on the CE router, include the unit statement:

```
unit logical-unit-number {
  tunnel {
    source address;
    destination address;
  }
  family inet {
    address address;
  }
}
```

You can include the unit statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]

Configuring GRE Tunnels Dynamically

When the router receives a VPN route to a BGP next-hop address but no MPLS path is available, a GRE tunnel can be dynamically generated to carry the VPN traffic across the BGP network. The GRE tunnel is instantiated into the inet.3 routing table.



NOTE: IPv4 routes are the only type of routes supported for dynamic GRE tunnels. Also, the routing platform must have a tunnel PIC.

To generate GRE tunnels dynamically, include the `dynamic-tunnels` statement:

```
dynamic-tunnels tunnel-name {
  destination-networks prefix;
  source-address address;
  tunnel-type gre;
}
```

You can include this statement at the following hierarchy levels:

- [edit routing-options]
- [edit logical-routers *logical-router-name* routing-options]

Specify the IPv4 prefix range (for example, 10/8 or 11.1/16) for the destination network by including the `destination-networks` statement. Only tunnels within the specified IPv4 prefix range are allowed to be initiated.

```
destination-networks prefix;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-routers *logical-router-name* routing-options dynamic-tunnels *tunnel-name*]

Specify the source address for the GRE tunnels by including the `source-address` statement. The source address specifies the address used as the source for the local tunnel endpoint. This could be any local address on the router (typically the router ID or the loopback address).

```
source-address address;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-routers *logical-router-name* routing-options dynamic-tunnels *tunnel-name*]

Specify the type of tunnel to be dynamically created by including the `tunnel-type` statement. The only currently valid value is `gre` (for GRE tunnels).

```
tunnel-type gre;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-routers *logical-router-name* routing-options dynamic-tunnels *tunnel-name*]

Configuring an ES Tunnel Interface for Layer 3 VPNs

An ES tunnel interface allows you to configure an IP Security (IPSec) tunnel between the PE and CE routers of a Layer 3 VPN. The IPSec tunnel can include one or more hops.

The following sections explain how to configure an ES tunnel interface between the PE and CE routers of a Layer 3 VPN:

- Configuring the ES Tunnel Interface on the PE Router on page 179
- Configuring the ES Tunnel Interface on the CE Router on page 180

Configuring the ES Tunnel Interface on the PE Router

To configure the ES tunnel interface on the PE router, include the unit statement:

```
unit logical-unit-number {
    tunnel {
        source source-address;
        destination destination-address;
    }
    family inet {
        address address;
        ipsec-sa security-association-name;
    }
}
```

You can include the unit statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]

By default, the tunnel destination address is assumed to be in the default Internet routing table, `inet.0`. For IPsec tunnels using manual security association (SA), if the tunnel destination address is not in the default `inet.0` routing table, you need to specify which routing table to search for the tunnel destination address by configuring the `routing-instance` statement. This is the case if the tunnel encapsulating interface is also configured under the routing instance.

```
unit logical-unit-number {
    tunnel {
        source address;
        destination address;
        routing-instance {
            destination routing-instance-name;
        }
    }
    family inet {
        address address;
        ipsec-sa security-association-name;
    }
    family mpls;
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]



NOTE: For IPsec tunnels using dynamic SA, the tunnel destination address must be in the default Internet routing table, `inet.0`.

To complete the ES tunnel interface configuration, include the `interface` statement for the ES interface under the appropriate routing instance:

```
interface interface-name;
```

You can include the `interface` statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

Configuring the ES Tunnel Interface on the CE Router

To configure the ES tunnel interface on the CE router, include the `unit` statement:

```
unit 0 {
  tunnel {
    source address;
    destination address;
  }
  family inet {
    address address;
    ipsec-sa security-association-name;
  }
}
```

You can include the `unit` statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-routers *logical-router-name* interfaces *interface-name*]

For more information about how to configure tunnel interfaces, see the *JUNOS Services Interfaces Configuration Guide*.

For more information about how to configure IPsec interfaces, see the *JUNOS System Basics Configuration Guide*.

Configuring IPsec Instead of MPLS Between PE Routers

A conventional Layer 3 BGP/MPLS VPN requires the configuration of MPLS label-switched paths (LSPs) between the PE routers. When a PE router receives a packet from a CE router, it performs a lookup in a specific VRF table for the IP destination address and obtains a corresponding MPLS label stack. The label stack is used to forward the packet to the egress PE router, where the bottom label is removed and the packet is forwarded to the specified CE router.

You can provide Layer 3 BGP/MPLS VPN service without an MPLS backbone. Instead of configuring MPLS LSPs between the PE routers, you configure GRE and IPsec tunnels between the PE routers. The MPLS information for the VPN (the VPN label) is encapsulated within an IP header and an IPsec header. The source address of the IP header is the address of the ingress PE router. The destination address has the BGP next hop, the address of the egress PE router.



NOTE: The IPsec tunnel requires the use of an ES PIC. The GRE tunnel requires the use of a Tunnel Services PIC.

To configure IPsec between PE routers, follow these steps:

1. Configure an IPsec tunnel between the PE routers. The source address is that of the ingress PE router, and the destination address is that of the egress PE router:

```

es-interface-name {
  unit unit-number {
    tunnel {
      source source-address;
      destination destination-address;
    }
    family inet {
      ipsec-sa sa-esp-dynamic;
      address address;
    }
    family mpls;
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
 - [edit logical-routers *logical-router-name* interfaces]
2. Configure IPsec on the PE router. For information about how to configure IPsec, see the *JUNOS System Basics Configuration Guide*.

3. Configure a GRE tunnel between the PE routers. Again, the source address is that of the ingress PE router, and the destination address is that of the egress PE router:

```

gr-interface-name {
  unit unit-number {
    family inet {
      address address;
    }
    family mpls;
    tunnel {
      source source-address;
      destination destination-address;
    }
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-routers *logical-router-name* interfaces]

4. Configure BGP between the PE routers:

```

bgp {
  group pe {
    type internal;
    local-address local-address;
    family inet {
      unicast;
    }
    family inet-vpn {
      unicast;
    }
    peer-as as-number;
    neighbor address;
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit protocols]
- [edit logical-routers *logical-router-name* protocols]

5. Configure the routing instance:

```

instance-type vrf;
interface interface-name;
route-distinguisher address;
vrf-import import-policy-name;
vrf-export export-policy-name;
protocols {
  bgp {
    group routing-instance-name {
      type external;
      peer-as as-number;
      as-override;
      neighbor address;
    }
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-routers *logical-router-name* routing-instances *routing-instance-name*]

6. Configure the policy options:

```

policy-statement import-policy-name {
  term 1 {
    from {
      protocol bgp;
      community community-name;
    }
    then accept;
  }
  term 2 {
    then reject;
  }
}
policy-statement export-policy-name {
  term 1 {
    from protocol [ bgp direct ];
    then {
      community add community-name;
      accept;
    }
  }
  term 2 {
    then reject;
  }
}
community community-name members target:target;

```

You can include these statements at the following hierarchy levels:

- [edit policy-options]
- [edit logical-routers *logical-router-name* policy-options]

7. Configure routing table groups to enable VPN route resolution in the inet.3 routing table:

```

interface-routes {
  rib-group inet if-rib;
}
rib inet.3 {
  static {
    route BGP-address-for-remote-PE next-hop gre-interface-name;
  }
}
rib-groups {
  if-rib {
    import-rib [ inet.0 inet.3 ];
  }
}

```

You can include these statements at the following hierarchy levels:

- [edit routing-options]
- [edit logical-routers *logical-router-name* routing-options]

Configuring SCU and DCU for Layer 3 VPNs

For information on how to configure source class usage (SCU) for a Layer 3 VPN loopback interface, see the *JUNOS Network Management Configuration Guide*.

For information on how to configure SCU and destination class usage (DCU) to count packets on Layer 3 VPNs, see the *JUNOS Network Interfaces Configuration Guide*.

Protocol-Independent Load Balancing for Layer 3 VPNs

Protocol-independent load balancing for Layer 3 VPNs allows the forwarding next hops of both the active route and alternative paths to be used for load balancing. Protocol-independent load balancing works in conjunction with Layer 3 VPNs. It supports the load balancing of VPN routes independently of the assigned route distinguisher. When protocol-independent load balancing is enabled, both routes to other PE routers and routes to directly connected CE routers are load-balanced.

When load-balancing information is created for a given route, the active path is marked as **Routing Use Only** in the output of the `show route table` command.

The following sections describe how to configure protocol-independent load balancing and how this configuration can affect routing policies:

- Configuring Load Balancing for Layer 3 VPNs on page 185
- Configuring Load Balancing and Routing Policies on page 186

Configuring Load Balancing for Layer 3 VPNs

To configure protocol-independent load balancing for Layer 3 VPNs, include the `multipath` statement:

```
multipath {
  vpn-unequal-cost;
}
```

If you include the `multipath` statement at the following hierarchy levels, protocol-independent load balancing is applied to the default routing table for that routing instance (*routing-instance-name.inet.0*):

- [edit routing-instances *routing-instance-name* routing-options]
- [edit logical routers *logical-router-name* routing-instances *routing-instance-name* routing-options]

If you include the `multipath` statement at the following hierarchy levels, protocol-independent load balancing is applied to the specified routing table:

- [edit routing-instances *routing-instance-name* routing-options rib *routing-table-name*]
- [edit logical routers *logical-router-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name*]

The `vpn-unequal-cost` statement is optional:

- If you do not configure the `vpn-unequal-cost` statement, protocol-independent load balancing is applied to VPN routes that are equal until the router identifier with regard to route selection.
- If you configure the `vpn-unequal-cost` statement, protocol-independent load balancing is applied to VPN routes that are equal until the IGP metric with regard to route selection.

Configuring Load Balancing and Routing Policies

If you enable protocol-independent load balancing for Layer 3 VPNs by including the `multipath` statement and if you also include the `load-balance per-packet` statement in the routing policy configuration, packets are not load-balanced.

For example, a PE router has the following VRF routing instance configured:

```
[edit routing-instances]
load-balance-example {
  instance-type vrf;
  interface fe-0/1/1.0;
  interface fe-0/1/1.1;
  route-distinguisher 2222:2;
  vrf-target target:2222:2;
  routing-options {
    multipath;
  }
  protocols {
    bgp {
      group group-example {
        import import-policy;
        family inet {
          unicast;
        }
        export export-policy;
        peer-as 4444;
        local-as 3333;
        multipath;
        as-override;
        neighbor 10.12.33.22;
      }
    }
  }
}
```

The PE router also has the following policy statement configured:

```
[edit policy-options policy-statement export-policy]
from protocol bgp;
then {
  load-balance per-packet;
}
```

When you include the `multipath` statement in the VRF routing instance configuration, the paths are no longer marked as BGP paths but are instead marked as multipath paths. Packets from the PE router are not load-balanced.

To ensure that VPN load-balancing functions as expected, do not include the `from protocol` statement in the policy statement configuration. The policy statement should be configured as follows:

```
[edit policy-options policy-statement export-policy]
then {
  load-balance per-packet;
}
```

For more information on how to configure per-packet load balancing, see the *JUNOS Policy Framework Configuration Guide*.

Configuring Layer 3 VPN Policing on Interfaces

You can use policing to control the amount of traffic flowing over the interfaces servicing a Layer 3 VPN. If policing is disabled on an interface, all the available bandwidth on a Layer 3 VPN tunnel can be used by a single CCC or TCC interface.

For more information about the `policer` statement, see the *JUNOS Policy Framework Configuration Guide*.

To enable Layer 3 VPN policing on an interface, include the `policer` statement:

```
policer {
  input policer-template-name;
  output policer-template-name;
}
```

If you configure CCC encapsulation, you can include the `policer` statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family ccc]
- [edit logical-routers *logical-router-name* interfaces *interface-name* unit *logical-unit-number* family ccc]

If you configure TCC encapsulation, you can include the `policer` statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family tcc]
- [edit logical-routers *logical-router-name* interfaces *interface-name* unit *logical-unit-number* family tcc]

Sending RADIUS Messages Through a Layer 3 VPN

You can send RADIUS messages through a Layer 3 VPN routing instance to customer RADIUS servers in a private network. To configure, include the `routing-instance` statement at the [edit access profile *profile-name* radius-server] hierarchy level and apply the profile to an interface with the `access-profile` statement at the [edit interfaces *interface-name* unit *logical-unit-number* ppp-options chap] hierarchy level. For more information, see the *JUNOS System Basics Configuration Guide*.