

Chapter 14

Configure LDP

The Label Distribution Protocol (LDP) is instance-aware. It can be configured for the master instance at the [edit protocols ldp] hierarchy level, or for a routing instance at the [edit routing instances routing-instance-name protocols] hierarchy level.

```
[edit]
ldp {
  explicit-null;
  (deaggregate | no-deaggregate);
  egress-policy policy-name;
  export [ policy-name ];
  import [ policy-name ];
  graceful-restart {
    disable;
    helper-disable;
    maximum-recovery-time value;
    recovery-time value;
  }
  interface interface-name {
    apply-groups;
    disable;
    hello-interval seconds;
    hold-time seconds;
    transport-address (interface | loopback);
  }
  keepalive-interval seconds;
  keepalive-timeout seconds;
  no-forwarding;
  preference preference;
  session address {
    authentication-key authentication-key;
  }
  traceoptions {
    apply-groups;
    file filename <replace> <size size> <files number> <no-stamp>
      <(world-readable | no-world-readable)>;
    flag flag <flag-modifier> <disable>;
  }
  track-igp-metric;
  traffic-statistics {
    file filename <replace> <size size> <files number>
      <(world-readable | no-world-readable)>;
    interval interval;
  }
  transport-address (router-id | loopback);
}
```

By default, LDP is disabled.

This chapter describes the minimum required LDP configuration and discusses the following configuration tasks:

- Minimum LDP Configuration on page 199
- Enable LDP on page 199
- Configure the LDP Hello Interval on page 199
- Configure the LDP Hold Time on page 200
- Configure the LDP Keepalive Interval on page 200
- Configure the LDP Keepalive Timeout on page 200
- Configure LDP Route Preferences on page 200
- Configure LDP Ultimate-Hop Popping on page 201
- Configure LDP Graceful Restart on page 201
- Configure LDP Received Label Filtering on page 203
- Configure LDP Outbound Label Filtering on page 205
- Enable LDP over RSVP-Established LSPs on page 207
- Configure LDP Transport Address Control on page 208
- Configure the LDP Egress Policy on page 208
- Configure FEC Deaggregation on page 209
- Configure the TCP MD5 Signature for LDP Session on page 210
- Trace LDP Protocol Traffic on page 210
- Collect LDP Statistics on page 212
- Configure LDP to Use the IGP Route Metric on page 214
- Configure the no-forwarding Statement on page 214
- Multiple-Instance LDP and Carrier-of-Carriers VPNs on page 214

For an LDP configuration example, see “Example: LDP Configuration” on page 212.

Minimum LDP Configuration

To enable LDP on all interfaces, include the following statement in the configuration file. All other LDP configuration statements are optional.

```
[edit]
protocols {
  ldp {
    interface all;
  }
}
```

Enable LDP

LDP is routing-instance-aware. To enable LDP on a specific interface, include the following statements at the [edit protocols ldp] hierarchy level or the [edit routing-instances *routing-instance-name* protocols ldp] hierarchy level:

```
[edit]
ldp {
  interface interface-name;
}
```

To enable LDP on all interfaces, specify *all* for *interface-name*.

If you have configured interface properties on a group of interfaces and want to disable LDP on one of the interfaces, include the disable statement within the LDP interface statement:

```
[edit]
protocols {
  ldp {
    ldp interface-name {
      disable;
    }
  }
}
```

Configure the LDP Hello Interval

LDP hello messages enable LDP nodes to discover one another and to detect the failure of a neighbor or of the link to the neighbor. Hello messages are sent periodically on all interfaces where LDP is enabled.

By default, LDP sends hello messages every 5 seconds. To modify how often LDP sends hello packets, include the hello-interval statement at the [edit protocols ldp interface *interface-name*] hierarchy level:

```
[edit protocols ldp interface interface-name]
hello-interval seconds;
```

Configure the LDP Hold Time

The hold time determines how long an LDP node should wait for a hello message before declaring a neighbor to be down. This value is sent as part of a hello message so that each LDP node tells its neighbors how long to wait. The values sent by each neighbor do not have to match. The hold time should be at least three times the hello interval. The default is 15 seconds. To modify the hold time, include the `hold-time` statement at the [edit protocols ldp interface *interface-name*] hierarchy level:

```
[edit protocols ldp interface interface-name]  
hold-time seconds;
```

Configure the LDP Keepalive Interval

The keepalive interval determines how often a message is sent over the session to ensure that the keepalive timeout is not exceeded. If no other LDP traffic is sent over the session in this much time, a keepalive message is sent. The default is 10 seconds. To modify the keepalive interval, include the `keepalive-interval` statement at the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level:

```
[edit]  
keepalive-interval seconds;
```

Configure the LDP Keepalive Timeout

After an LDP session is established, messages must be exchanged periodically to ensure that the session is still working. The keepalive timeout defines the amount of time that the neighbor LDP node waits before deciding that the session has failed. This value is usually set to at least three times the keepalive interval. The default is 30 seconds. To modify the keepalive interval, include the `keepalive-timeout` statement at the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level:

```
[edit]  
keepalive-timeout seconds;
```

Configure LDP Route Preferences

When several protocols calculate routes to the same destination, route preferences are used to select which route is installed in the forwarding table. The route with the lowest preference value is selected. The preference value can be a number in the range 0 through 255. By default, LDP routes have a preference value of 9. To modify the route preferences, include the `preference` statement at the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level:

```
[edit]  
preference preference;
```

Configure LDP Ultimate-Hop Popping

You can control the label value advertised on the egress router of an LSP. The default advertised label is Label 3 (Implicit Null Label). If Label 3 is advertised, the penultimate hop router removes the label and sends the packet to the egress router. By enabling ultimate hop popping, Label 0 (IPv4 Explicit Null Label) is advertised. Ultimate-hop popping ensures that any packets traversing an MPLS network include a label.

To configure ultimate hop popping, include the explicit-null statement at the [edit protocols ldp] hierarchy level:

```
[edit protocols ldp]
explicit-null;
```



Juniper Networks routers queue packets based on the incoming label. Routers from other vendors might queue packets differently. Keep this in mind when working with networks containing routers from multiple vendors.

Note

For more information about labels, see “Label Description” on page 20 and “Label Allocation” on page 21.

Configure LDP Graceful Restart

By default, graceful restart helper mode is enabled, but graceful restart is disabled. Thus, the default behavior of a router is to assist neighboring routers attempting a graceful restart, but not to attempt a graceful restart itself.

To configure LDP graceful restart, complete the procedures in the following sections:

Enable Graceful Restart on page 201

Disable LDP Graceful Restart or Helper Mode on page 202

Configure Recovery Time and Maximum Recovery Time on page 203

Enable Graceful Restart

To enable LDP graceful restart, you also need to enable graceful restart on the router. To enable graceful restart, include the graceful-restart statement at the [edit routing-options] hierarchy level:

```
[edit]
routing-options {
  graceful-restart;
}
```

The graceful-restart statement enables graceful restart for all protocols supporting this feature on the router. For more information about graceful restart, see the *JUNOS Internet Software Configuration Guide: Routing and Routing Protocols*.

By default, LDP graceful restart is enabled when you enable graceful restart at both the LDP protocol level and on all the routing instances. However, you can disable both LDP graceful restart and LDP graceful restart helper mode.

Disable LDP Graceful Restart or Helper Mode

To disable LDP graceful restart and recovery, include the disable statement at the [edit protocols ldp graceful-restart] hierarchy level or at the [edit protocols routing-instances routing-instance-name routing-options graceful-restart] hierarchy level:

```
[edit]
ldp {
  graceful-restart {
    disable;
  }
}
```

You can disable helper mode at the LDP protocols level only. You cannot disable helper mode for a specific routing instance. To disable LDP helper mode, include the helper-disable statement at the [edit protocols ldp graceful-restart] hierarchy level:

```
[edit protocols]
ldp {
  graceful-restart {
    helper-disable;
  }
}
```

The following LDP graceful restart configurations are possible:

LDP graceful restart and helper mode are both enabled.

LDP graceful restart is disabled but helper mode is enabled. A router configured in this way cannot restart gracefully, but can help a restarting neighbor.

LDP graceful restart and helper mode are both disabled. The router does not use LDP graceful restart or the graceful restart TLV sent in the initialization message. The router behaves as a router that cannot support LDP graceful restart.

A configuration error is issued if you attempt to enable graceful restart and disable helper mode.

Configure Recovery Time and Maximum Recovery Time

The recovery time is the amount of time a router waits for LDP to restart gracefully. The recovery time period begins when an initialization message is sent or received. This is also typically the amount of time that a neighboring router maintains its information about the restarting router, allowing it to continue to forward traffic.

To prevent a neighboring router from being adversely affected if it receives a false value for the recovery time from the restarting router, you can configure the maximum recovery time on the neighboring router. A neighboring router maintains its state for the shorter of the two times. For example, router A is performing an LDP graceful restart. It has sent a recovery time of 900 seconds to neighboring router B. However, router B has its maximum recovery time configured at 400 seconds. Router B will only wait for 400 seconds before it purges its LDP information from router A.

Configure the `recovery-time` statement and the `maximum-recovery-time` statement at the `[edit protocols ldp graceful-restart]` hierarchy level:

```
[edit protocols]
  ldp {
    graceful-restart {
      maximum-recovery-time seconds;
      recovery-time seconds;
    }
  }
}
```

Configure LDP Received Label Filtering

You can filter received LDP label bindings, applying policies to accept or deny bindings advertised by neighboring routers. To configure received label filtering, include the `import` statement at the `[edit protocols ldp]` hierarchy level or the `[edit routing instances routing-instance-name protocols ldp]` hierarchy level.

```
[edit]
  import [policy-name];
```

The named policy (configured at the `[edit policy-options]` hierarchy level) is applied to all label bindings received from all LDP neighbors. All filtering is done using from statements. Table 4 lists the only from operators that apply to LDP received label filtering.

Table 4: from Operators That Apply to LDP Received Label Filtering

from Operator	Description
interface	Matches on bindings received from a neighbor that is adjacent over the specified interface.
neighbor	Matches on bindings received from the specified LDP router ID.
next-hop	Matches on bindings received from a neighbor advertising the specified interface address.
route-filter	Matches on bindings with the specified prefix.

If a binding is filtered, it still appears in the LDP database, but is not considered for installation as part of a label-switched path.

Generally, applying policies in LDP can only be used to block the establishment of LSPs, not to control their routing. This is because the path that an LSP follows is determined by unicast routing, and not by LDP. However, when there are multiple equal-cost paths to the destination through different neighbors, you can use LDP filtering to exclude some of the possible next-hops from consideration. (Otherwise, LDP chooses one of the possible next-hops at random.)

LDP sessions are not bound to interfaces or interface addresses. LDP advertises only per-router (not per-interface) labels, so if multiple parallel links exist between two routers, only one LDP session is established and it is not bound to a single interface. Be careful, when a router has multiple adjacencies to the same neighbor, to ensure that the filter does what is expected. (Generally, using next-hop and interface is not appropriate in this case.)

If a label has been filtered (meaning that it has been rejected by the policy and is not used to construct an LSP), it is marked as filtered in the database:

```
user@host> show ldp database
Input label database, 10.10.255.1:0-10.10.255.6:0
Label Prefix
3 10.10.255.6/32 (Filtered)
Output label database, 10.10.255.1:0-10.10.255.6:0
Label Prefix
3 10.10.255.1/32 (Filtered)
```

For more information on how to configure policies for LDP, see the *JUNOS Internet Software Configuration Guide: Policy Framework*.

Examples: Configure Received Label Filtering

Accept only /32 prefixes from all neighbors:

```
[edit]
protocols {
  ldp {
    import only-32;
    ...
  }
}
policy-options {
  policy-statement only-32 {
    term first {
      from {
        route-filter 0.0.0.0/0 upto /31;
      }
      then reject;
    }
    then accept;
  }
}
```

Accept 131.108/16 from router ID 10.10.255.2 and accept all prefixes from all other neighbors:

```
[edit]
protocols {
  ldp {
    import nosy-neighbor;
    ...
  }
}
policy-options {
  policy-statement nosy-neighbor {
    term first {
      from {
        neighbor 10.10.255.2;
        route-filter 131.108.0.0/16 orlonger accept;
        route-filter 0.0.0.0/0 orlonger reject;
      }
    }
  }
  then accept;
}
}
```

Configure LDP Outbound Label Filtering

You can configure export policies to filter LDP outbound labels. You can filter outbound label bindings by applying routing policies to block bindings from being advertised to neighboring routers. To configure outbound label filtering, include the export statement at the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level.

```
[edit]
export [ policy-name ];
```

The named export policy (configured at the [edit policy-options] hierarchy level) is applied to all label bindings transmitted to all LDP neighbors. The only from operator that applies to LDP outbound label filtering is route-filter, which matches bindings with the specified prefix. The only to operators that apply to outbound label filtering are the operators in Table 5.

Table 5: to Operators for LDP Outbound Label Filtering

to Operator	Description
interface	Matches on bindings sent to a neighbor that is adjacent over the specified interface
neighbor	Matches on bindings sent to the specified LDP router ID
next-hop	Matches on bindings sent to a neighbor advertising the specified interface address

If a binding is filtered, the binding is not advertised to the neighboring router, but it can be installed as part of a label-switched path on the local router. You can apply policies in LDP to block the establishment of LSPs, but not to control their routing. The path an LSP follows is determined by unicast routing, not by LDP.

LDP sessions are not bound to interfaces or interface addresses. LDP advertises only per-router (not per-interface) labels. If multiple parallel links exist between two routers, only one LDP session is established and it is not bound to a single interface.

Do not use the next-hop and interface operators when a router has multiple adjacencies to the same neighbor.

Filtered labels are marked in the database:

```
user@host> show ldp database
Input label database, 10.10.255.1:0-10.10.255.3:0
Label Prefix
100007 10.10.255.2/32
3 10.10.255.3/32
Output label database, 10.10.255.1:0-10.10.255.3:0
Label Prefix
3 10.10.255.1/32
100001 10.10.255.6/32 (Filtered)
```

For more information on how to configure policies for LDP, see the *JUNOS Internet Software Configuration Guide: Policy Framework*.

Examples: Configure Outbound Label Filtering

Block transmission of 10.10.255.6/32 to all neighbors:

```
[edit protocols]
ldp {
  export block-one;
}
policy-options {
  policy-statement block-one {
    term first {
      from {
        route-filter 10.10.255.6/32 exact;
      }
      then reject;
    }
    then accept;
  }
}
```

Send only 131.108/16 to router ID 10.10.255.2, and send all prefixes to all other routers:

```
[edit protocols]
ldp {
  export limit-lsps;
}
policy-options {
  policy-statement limit-lsps {
    term allow-one {
      from {
        route-filter 131.108.0.0/16 orlonger;
      }
      to {
        neighbor 10.10.255.2;
      }
      then accept;
    }
    term block-the-rest {
      to {
        neighbor 10.10.255.2;
      }
      then reject;
    }
    then accept;
  }
}
```

Enable LDP over RSVP-Established LSPs

You can run LDP over LSPs established by RSVP, effectively tunneling the LDP-established LSP through the one established by RSVP. To do so, you must enable LDP on the lo0.0 interface (see “Enable LDP” on page 199). Additionally, you must configure the LSPs over which you want LDP to operate by including the `ldp-tunneling` statement at the `[edit protocols mpls label-switched-path lsp-path-name]` hierarchy level:

```
[edit]
protocols {
  mpls {
    label-switched-path lsp-path-name {
      from source;
      to destination;
      ldp-tunneling;
    }
  }
}
```

For more information about tunneling LDP LSPs, see “Tunnel LDP LSPs in RSVP LSPs” on page 191.

Configure LDP Transport Address Control

You can control the transport address used by LDP. The transport address is the address used for the TCP session over which LDP is running. To configure transport address control, include the transport-address statement:

```
transport-address ( router-id | interface );
```

You can configure the transport address globally for all LDP sessions (at the [edit protocols ldp] hierarchy level), for each LDP instance (at the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level), or for each interface (at either the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level).

If you select router-id, the address of the router identifier is used as the transport address (unless otherwise configured, the router identifier is typically the same as the loopback address). If you select interface, the interface address is used as the transport address for any LDP sessions to neighbors that can be reached over that interface. Note that the router identifier is used as the transport address by default.

You cannot use transport-address interface when there are multiple parallel links to the same LDP neighbor, because the LDP specification requires that the same transport address be advertised on all interfaces to the same neighbor. If LDP detects multiple parallel links to the same neighbor, it disables interfaces to that neighbor one by one until the condition is cleared, either by disconnecting the neighbor on an interface or by configuring transport-address router-id.

Configure the LDP Egress Policy

You can control the set of prefixes that are advertised into LDP and cause the router to be the egress router for those prefixes. By default, only the loopback address is advertised into LDP. To configure the set of prefixes from the routing table to be advertised into LDP, include the egress-policy statement at the [edit protocols ldp] hierarchy level or the [edit routing instances *routing-instance-name* protocols ldp] hierarchy level:

```
egress-policy policy-name;
```



Note

If you configure an egress policy for LDP that does not include the loopback address, it is no longer advertised in LDP. To continue to advertise the loopback address, you need to explicitly configure it as a part of the LDP egress policy.

The named policy (configured at the [edit policy-options] hierarchy level) is applied to all routes in the routing table. Those routes that match the policy are advertised into LDP. You can control the set of neighbors to which those prefixes are advertised using the export statement. Only from operators are considered; you can use any valid from operator. For more information, see the *JUNOS Internet Software Configuration Guide: Routing and Routing Protocols*.

Example: Configure the LDP Egress Policy

Advertise all connected routes into LDP:

```
[edit protocols]
ldp {
  egress-policy connected-only;
}
policy-options {
  policy-statement connected-only {
    from {
      protocol direct;
    }
    then accept;
  }
}
```

Configure FEC Deaggregation

When an LDP egress router advertises multiple prefixes, the prefixes are bound to a single label and aggregated into a single Forwarding Equivalence Class (FEC). By default, LDP maintains this aggregation as the advertisement traverses the network.

By default, because an LSP cannot be split across multiple next-hops and all the prefixes are bound into a single LSP, you cannot load-balance across equal-cost paths.

To change the default to load-balance across equal-cost paths, deaggregate FECs. Deaggregating FECs causes each prefix to be bound to a separate label and become a separate LSP.

To configure deaggregated FECs, include the `deaggregate` statement at the `[edit protocols ldp]` hierarchy level or the `[edit routing instances routing-instance-name protocols ldp]` hierarchy level:

```
deaggregate;
```

For all LDP sessions, you can only configure deaggregated FECs globally.

Deaggregating an FEC allows the resulting multiple LSPs to be distributed across multiple equal-cost paths and distributes LSPs across the multiple next-hops on the egress segments but installs only one next-hop per LSP.

To aggregate FECs, include the `no-deaggregate` statement at the `[edit protocols ldp]` hierarchy level or the `[edit routing instances routing-instance-name protocols ldp]` hierarchy level:

```
no-deaggregate;
```

For all LDP sessions, you can only configure aggregated FECs globally.

Configure the TCP MD5 Signature for LDP Session

You can configure an MD5 signature for an LDP TCP connection to protect against the introduction of spoofed TCP segments into LDP session connection streams.

A router using the MD5 signature option is configured with a password for each peer for which authentication is required. The password is stored encrypted.

LDP hello adjacencies can still be created even when peering interfaces are configured with different security signatures. However, the TCP session cannot be authenticated and is never established.



Note

If you apply an MD5 signature to an LDP interface with an established session, it drops the TCP connection and all the associated label bindings to the FEC entries for that session. The session regenerates the database information for that session once both interfaces agree on a common security method and password.

To configure an MD5 signature for an LDP TCP connection, include the following statements at the [edit protocols ldp] hierarchy level:

```
[edit protocols ldp]
session address {
  authentication-key authentication-key;
}
```

Use the session statement to configure the address for the remote end of the LDP session.

The *authentication-key* (password) can be up to 69 characters long. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks.

Trace LDP Protocol Traffic

To trace LDP protocol traffic, you can specify options in the global traceoptions statement at the [edit routing-options] hierarchy level, and you can specify LDP-specific options by including the traceoptions statement at either the [edit protocols ldp] hierarchy level or the [edit routing instances routing-instance-name protocols ldp] hierarchy level:

```
[edit]
traceoptions {
  file filename <replace> <size size> <files number> <no-stamp>
    <(world-readable | no-world-readable)>;
  flag flag <flag-modifier> <disable>;
}
```

Use the file statement to specify the name of the file that receives the output of the tracing operation. All files are placed in the directory /var/log. We recommend that you place LDP tracing output in the file ldp-log.

The following trace flags display the operations associated with the sending and receiving of various LDP messages. Each can carry one or more of the following modifiers:

address—Trace the operation of address and address withdrawal messages.

binding—Trace label-binding operations.

error—Trace error conditions.

event—Trace protocol events.

initialization—Trace the operation of initialization messages.

label—Trace the operation of label request, label map, label withdrawal, and label release messages.

notification—Trace the operation of notification messages.

packet—Trace the operation of address, address withdrawal, initialization, label request, label map, label withdrawal, label release, notification, and periodic messages. This modifier is equivalent to setting the address, initialization, label, notification, and periodic modifiers.

path—Trace label-switched path operations.

periodic—Trace the operation of hello and keepalive messages.

state—Trace protocol state transitions.

Examples: Trace LDP Protocol Traffic

Trace LDP Path messages in detail:

```
[edit]
protocols {
  ldp {
    traceoptions {
      file ldp size 10m files 5;
      flag path;
    }
  }
}
```

Trace all LDP outgoing messages:

```
[edit]
protocols {
  ldp {
    traceoptions {
      file ldp size 10m files 5;
      flag packets;
    }
  }
}
```

Trace all LDP error conditions:

```
[edit]
protocols {
  ldp {
    traceoptions {
      file ldp size 10m files 5;
      flag error;
    }
  }
}
```

Example: LDP Configuration

The following shows an example of an LDP configuration:

```
[edit]
protocols {
  ldp {
    traceoptions {
      file ldp size 10m files 5 world-readable;
      flag packets receive;
      flag binding;
    }
    interface all {
    }
  }
}
```

Collect LDP Statistics

LDP traffic statistics show the volume of traffic that has passed through a particular forwarding equivalence class (FEC) on a router.

When you configure the traffic-statistics statement at the [edit protocols ldp] hierarchy level, the LDP traffic statistics are gathered periodically and written to a file. You can configure how often statistics are collected (in seconds) using the interval option. The default collection interval is five minutes. You must configure an LDP statistics file; otherwise LDP traffic statistics are not gathered. If the LSP goes down, the LDP statistics are reset.

To collect LDP traffic statistics, configure the following statements at the [edit protocols ldp] hierarchy level:

```
[edit protocols ldp]
traffic-statistics {
  file filename <files number> <replace> <size size>
    <(world-readable | no-world-readable)>;
  interval interval;
}
```

This section includes the following topics:

LDP Statistics Output on page 213

LDP Statistics Limitations on page 213

LDP Statistics Output

Following is sample of output from an LDP statistics file:

FEC	Type	Packets	Bytes	Shared
10.255.350.448/32	Transit	0	0	No
	Ingress	0	0	No
10.255.350.450/32	Transit	0	0	Yes
	Ingress	0	0	No
10.255.350.451/32	Transit	0	0	No
	Ingress	0	0	No
220.220.220.1/32	Transit	0	0	Yes
	Ingress	0	0	No
220.220.220.2/32	Transit	0	0	Yes
	Ingress	0	0	No
220.220.220.3/32	Transit	0	0	Yes
	Ingress	0	0	No

May 28 15:02:05, read 12 statistics in 00:00:00 seconds

The following describes each of the columns of data collected in the LDP statistics file:

Bytes—The number of bytes of data passed by the FEC since its LSP came up.

FEC—The forwarding equivalence class (FEC) for which LDP traffic statistics are collected.

Packets—The number of packets passed by the FEC since its LSP came up.

read—This number (appears next to the date and time) might differ from the actual number of the statistics displayed. Some of the statistics are summarized before being displayed.

Shared—A Yes value indicates that several prefixes are bound to the same label (for example, when several prefixes are advertised with an egress policy). The LDP traffic statistics for this case apply to all the prefixes and should be treated as such.

Type—The type of traffic originating from a router is either Ingress traffic (originating from this router) or Transit traffic (forwarded through this router).

LDP Statistics Limitations

The following are issues related to collecting LDP statistics by configuring the traffic-statistics statement:

There is no way to clear the LDP statistics.

If you shorten the specified interval, a new LDP statistics request is issued only if the statistics timer expires later than the new interval.

A new LDP statistics collection operation cannot be started until the previous one has finished. If the interval is short or if the number of LDP statistics is large, the time gap between the two statistics collections might be longer than the interval.

When an LSP goes down, the LDP statistics are reset.

Configure LDP to Use the IGP Route Metric

Use the `track-igp-metric` statement if you want the IGP route metric to be used for the LDP routes instead of the default LDP route metric (the default LDP route metric is 1).

Configure the `track-igp-metric` statement at either the `[edit protocols ldp]` hierarchy level or the `[edit routing instances routing-instance-name protocols ldp]` hierarchy level:

```
track-igp-metric;
```

Configure the no-forwarding Statement

By configuring the `no-forwarding` statement, you can prevent ingress routes from being added to the `inet.0` routing table instead of the `inet.3` routing table even if you enabled the `traffic-engineering bgp-igp` statement at the `[edit protocols mpls]` hierarchy level. By default, the `no-forwarding` statement is disabled.

Configure the `no-forwarding` statement at the `[edit protocols ldp]` hierarchy level or at the `[edit routing instances routing-instance-name protocols ldp]` hierarchy level:

```
no-forwarding;
```

Multiple-Instance LDP and Carrier-of-Carriers VPNs

By configuring multiple LDP routing instances, you can use LDP to advertise labels in a Carrier of Carriers VPN from a core provider provider edge (PE) router to a customer carrier customer edge (CE) router. This is especially useful when the carrier customer is a basic Internet service provider (ISP) and wants to restrict full Internet routes to its PE routers. By using LDP instead of BGP, the carrier customer shields its other internal routers from the Internet. Multiple-instance LDP is also useful when a carrier customer wants to provide Layer 2 or Layer 3 VPN services to its customers.

For an example of how to configure multiple LDP routing instances for carrier-of-carriers VPNs, see the *JUNOS Internet Software Feature Guide* on the product documentation page of the Juniper Networks Web site, located at <http://www.juniper.net/>.