LDP Configuration Commands

Generic Commands

ldp

Syntax **[no] ldp**

Context config>router

Default This command creates the context to configure an LDP parameters. LDP is not enabled by default and must be explicitly enabled (**no shutdown**).

To suspend the LDP protocol, use the **shutdown** command. Configuration parameters are not affected.

The **no** form of the command deletes the LDP protocol instance, removing all associated configuration parameters. The LDP instance must first be disabled with the **shutdown** command before being deleted.

Default none (LDP must be explicitly enabled)

ldp-shortcut

Syntax **[no] ldp-shortcut**

Context config>router

Description This command enables the resolution of IGP routes using LDP LSP across all network interfaces participating in the IS-IS and OSPF routing protocol in the system.

When LDP shortcut is enabled, LDP populates the routing table with next-hop entries corresponding to all prefixes for which it activated an LDP FEC. For a given prefix, two route entries are populated in the system routing table. One corresponds to the LDP shortcut next-hop and has an owner of LDP. The other one is the regular IP next-hop. The LDP shortcut next-hop always has preference over the regular IP next-hop for forwarding user packets and specified control packets over a given outgoing interface to the route next-hop.

All user and specified control packets for which the longest prefix match in RTM yields the FEC prefix will be forwarded over the LDP LSP.

When an IPv4 packet is received on an ingress network interface, a subscriber IES interface, or a regular IES interface, the lookup of the packet by the ingress forwarding engine will result in the packet being sent labeled with the label stack corresponding to the NHLFE of the LDP LSP when the preferred RTM entry corresponds to an LDP shortcut.

If the preferred RTM entry corresponds to an IP next-hop, the IPv4 packet is forwarded without a label.

When ECMP is enabled and multiple equal-cost next-hops exit for the IGP route, the ingress forwarding engine will spray the packets for this route based on hashing routine currently supported for IPv4 packets. When the preferred RTM entry corresponds to an LDP shortcut route, spraying will be performed across the multiple next-hops for the LDP FEC. The FEC next-hops can either be direct link LDP neighbors or T-LDP neighbors reachable over RSVP LSPs in the case of LDP-over-RSVP but not both.

When the preferred RTM entry corresponds to a regular IP route, spraying will be performed across regular IP next-hops for the prefix.

The **no** form of this command disables the resolution of IGP routes using LDP shortcuts.

Default no ldp-shortcut

shutdown

Syntax [**no**] **shutdown**

Context config>router>ldp
config>router>ldp>if-params>interface
config>router>ldp>if-params>interface>ipv4
config>router>ldp>if-params>interface>ipv6
config>router>ldp>targ-session>peer
config>router>ldp>targeted-session>peer-template
config>router>ldp>egr-stats>fec-prefix
config>router>ldp>aggregate-prefix-match

Description This command administratively disables an entity. When disabled, an entity does not change, reset, or remove any configuration settings or statistics.

The operational state of the entity is disabled as well as the operational state of any entities contained within. Many objects must be shut down before they may be deleted. For an LDP interface, the **shutdown** command exists under the main interface context and under each of the interface IPv4 and IPv6 contexts.

- **shutdown** under the **interface** context brings down both IPv4 and IPv6 Hello adjacencies and stops Hello transmission in both contexts.
- **shutdown** under the **interface** IPv4 or IPv6 contexts brings down the Hello adjacency and stops Hello transmission in that context only.

The user can also delete the entire IPv4 or IPv6 context under the interface with the **no ipv4** or **no ipv6** command which in addition to bringing down the Hello adjacency will delete the configuration

The **no** form of this command administratively enables an entity.

Unlike other commands and parameters where the default state is not indicated in the configuration file, the **shutdown** and **no shutdown** states are always indicated in system generated configuration files.

Default no shutdown

aggregate-prefix-match

Syntax [**no**] **aggregate-prefix-match**

Context config>router>ldp

Description The command enables the use by LDP of the aggregate prefix match procedures.

When this option is enabled, LDP performs the following procedures for all prefixes. When an LSR receives a FEC-label binding from an LDP neighbor for a given specific FEC1 element, it will install the binding in the LDP FIB if:

- It is able to perform a successful longest IP match of the FEC prefix with an entry in the routing table, and
- The advertising LDP neighbor is the next-hop to reach the FEC prefix.

When such a FEC-label binding has been installed in the LDP FIB, then LDP programs an NHLFE entry in the egress data path to forward packets to FEC1. It also advertises a new FEC-label binding for FEC1 to all its LDP neighbors.

When a new prefix appears in the routing table, LDP inspects the LDP FIB to determine if this prefix is a better match (a more specific match) for any of the installed FEC elements. For any FEC for which this is true, LDP may have to update the NHLFE entry for this FEC.

When a prefix is removed from the routing table, LDP inspects the LDP FIB for all FEC elements which matched this prefix to determine if another match exists in the routing table. If so, it updates the NHLFE entry accordingly. If not, it sends a label withdraw message to its LDP neighbors to remove the binding.

When the next hop for a routing prefix changes, LDP updates the LDP FIB entry for the FEC elements which matched this prefix. It also updates the NHLFE entry for these FEC elements accordingly.

The **no** form of this command disables the use by LDP of the aggregate prefix procedures and deletes the configuration. LDP resumes performing exact prefix match for FEC elements.

Default no aggregate-prefix-match

prefix-exclude

Syntax **prefix-exclude** *policy-name* [*policy-name*...(up to 5 max)]
no prefix-exclude

Context config>router>ldp>aggregate-prefix-match

Description This command specifies the policy name containing the prefixes to be excluded from the aggregate prefix match procedures. In this case, LDP will perform an exact match of a specific FEC element prefix as opposed to a longest match of one or more LDP FEC element prefixes, against this prefix when it receives a FEC-label binding or when a change to this prefix occurs in the routing table.

The **no** form of this command removes all policies from the configuration.

Default no prefix-exclude.

class-forwarding

Syntax **class-forwarding**
no class-forwarding

Context config>router>ldp

Description This command enables Class-Based Forwarding (CBF) capability. When this command is enabled, LDP prefixes resolved to a set of ECMP tunnel next-hops will have their packets forwarded to the LSP which is configured to carry the forwarding class the packet was classified to. As a pre-requisite for that forwarding behavior to happen, the user must have enabled IGP shortcuts in the routing instance, enabled ecmp in the global routing instance and have enabled the advertisement of unicast prefix FECs on the Targeted LDP (T-LDP) session to the peer. The user must also have assigned forwarding classes to LSPs (see **config**>**router**>**mpls**>**lsp**>**class-forwarding**).

Class based forwarding will occur when a set of ECMP tunnel next hops is considered consistent from a CBF perspective. It is the case if at least one CBF configuration (**fc** or **default-lsp**) is assigned to one or more LSPs. If no LSP of the set has the **default-lsp** option assigned, one LSP will automatically be selected for that assignment by LDP (the one with the lowest tunnel-id within the set of LSPs with one or more forwarding classes assigned). Multiple LSPs can have a same forwarding class assigned. However, for each of these forwarding classes only a single LSP will be used to forward packets classified into this forwarding class. That LSP is the one with the lowest tunnel-id amongst those sharing a given forwarding class. Similarly, multiple LSPs can have the **default-lsp** configuration assigned. Only a single one will be designated to be the Default LSP. That LSP is the one with the lowest tunnel-id amongst those with the **default-lsp** configuration assigned.

Under normal conditions, LDP prefix packets will be sprayed over a set of ECMP tunnel next-hops by selecting either the LSP to which is assigned the forwarding class of the packets, if one exists, or the Default LSP, if one does not exist. However, If the IOM detects that the LSP to which is assigned a forwarding class is not usable, it will switch the forwarding of packets classified to that forwarding class into the Default LSP, and if the IOM detects that the Default LSP is not usable, then it will revert to regular ECMP spraying across all tunnels in the set of ECMP tunnel next-hops. In other words, the CBF is suspended until LDP downloads a new consistent set of tunnel next-hops for the FEC.

This command only applies to LSR forwarding LDP FEC prefix packets over a set of MPLS LSPs using IGP shortcuts. It does not apply to LER forwarding of shortcut packets over LDP FEC, which is resolved to a set of MPLS LSPs using IGP shortcuts, nor does it apply to LER forwarding of packets of VPRN and Layer-2 services, which use auto-binding to LDP when the LDP FEC is resolved to a set of MPLS LSPs using IGP shortcuts.

The **no** form of this command disables the class based forwarding capability. In that case, LDP prefixes resolved to a set of ECMP tunnel next-hops will have their packets forwarded according to ECMP spraying.

Default **no class-forwarding**

egress-statistics

Syntax **egress-statistics**

Context config>router>ldp

Description This command provides the context for the user to enter the LDP FEC prefix for the purpose of enabling egress data path statistics at the ingress LER for this FEC.

Default none

fec-prefix

Syntax [**no**] **fec-prefix** *ip-prefix[/mask]*

Context config>router>ldp>egr-stats

Description This command configures statistics in the egress data path at the ingress LER or LSR for an LDP FEC. The user must execute the **no shutdown** command for this command to effectively enable statistics. The egress data path counters will be updated for both originating and transit packets. Originating packets may be service packets or IP user and control packets forwarded over the LDP LSP when used as an IGP shortcut. Transit packets of the FEC which are label switched on this node.

When ECMP is enabled and multiple paths exist for a FEC, the same set of counters are updated for each packet forwarded over any of the NHLFEs associated with this FEC and for as long as this FEC is active.

The statistics can be enabled on prefix FECs imported from both LDP neighbors and T-LDP neighbors (LDP over RSVP).Only /32 FEC prefixes are accepted. Service FECs, i.e., FEC 128 and FEC 129 are not valid. LDP FEC egress statistics are not collected at the Penultimate-Popping Hop (PHP) node for a LDP FEC using an implicit null label.

The **no** form of this command disables the statistics in the egress data path and removes the accounting policy association from the LDP FEC.

Default none

accounting-policy

Syntax **accounting-policy** *acct-policy-id*
no accounting-policy

Context config>router>ldp>egr-stats

Description This command associates an accounting policy to the MPLS instance.

An accounting policy must be defined before it can be associated else an error message is generated.

The **no** form of this command removes the accounting policy association.

Default none

Parameters *acct-policy-id* — Enter the accounting *policy-id* as configured in the **config>log>accounting-policy** context.

Values 1 — 99

collect-stats

Syntax [**no**] **collect-stats**

Context config>router>ldp>egr-stats

Description This command enables accounting and statistical data collection. When applying accounting policies the data, by default, is collected in the appropriate records and written to the designated billing file.

When the **no collect-stats** command is issued the statistics are still accumulated by the forwarding engine. However, the CPU will not obtain the results and write them to the billing file. If a subsequent **collect-stats** command is issued then the counters written to the billing file include all the traffic while the **no collect-stats** command was in effect.

Default collect-stats

export

Syntax **export** *policy-name* [*policy-name* …upto 5 max]
no export

Context config>router>ldp

Description This command specifies the export route policies used to determine which routes are exported to LDP. Policies are configured in the **config>router>policy-options** context.

If no export policy is specified, non-LDP routes will not be exported from the routing table manager to LDP. LDP-learned routes will be exported to LDP neighbors. Present implementation of export policy (outbound filtering) can be used "only" to add FECs for label propagation. The export policy does not control propagation of FECs that an LSR receives from its neighbors.

If multiple policy names are specified, the policies are evaluated in the order they are specified. The first policy that matches is applied. If multiple export commands are issued, the last command entered will override the previous command. A maximum of 5 policy names can be specified.

The **no** form of the command removes all policies from the configuration.

Default **no export** — No export route policies specified.

Parameters *policy-name* — The export route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

The specified name(s) must already be defined.

fast-reroute

Syntax [**no**] **fast-reroute**

Context config>router>ldp

Description This command enables LDP Fast-Reroute (FRR) procedures. When enabled, LDP uses both the primary next-hop and LFA next-hop, when available, for resolving the next-hop of an LDP FEC against the corresponding prefix in the routing table. This will result in LDP programming a primary NHLFE and a backup NHLFE into the forwarding engine for each next-hop of a FEC prefix for the purpose of forwarding packets over the LDP FEC.

When any of the following events occurs, LDP instructs in the fast path the forwarding engines to enable the backup NHLFE for each FEC next-hop impacted by this event:

- An LDP interface goes operationally down, or is admin shutdown.
- An LDP session to a peer went down as the result of the Hello or Keep-Alive timer expiring.
- The TCP connection used by a link LDP session to a peer went down, due say to next-hop tracking of the LDP transport address in RTM, which brings down the LDP session.
- A BFD session, enabled on a T-LDP session to a peer, times-out and as a result the link LDP session to the same peer and which uses the same TCP connection as the T-LDP session goes also down.
- A BFD session enabled on the LDP interface to a directly connected peer, times out and brings down the link LDP session to this peer.

The **tunnel-down-dump-time** option or the **label-withdrawal-delay** option, when enabled, does not cause the corresponding timer to be activated for a FEC as long as a backup NHLFE is still available.

Note that because LDP can detect the loss of a neighbor/next-hop independently, it is possible that it switches to the LFA next-hop while IGP is still using the primary next-hop. Also, when the interface for the previous primary next-hop is restored, IGP may re-converge before LDP completed the FEC exchange with it neighbor over that interface. This may cause LDP to de-program the LFA next-hop

from the FEC and blackhole traffic. In order to avoid this situation, it is recommended to enable IGP-LDP synchronization on the LDP interface.

When the SPF computation determines there is more than one primary next-hop for a prefix, it will not program any LFA next-hop in RTM. Thus, the LDP FEC will resolve to the multiple primary next-hops that provide the required protection.

The **no** form of this command disables LDP FRR.

Default no fast-reroute

export-tunnel-table

Syntax [**no**] **export-tunnel-table** *policy-name*

Context config>router>ldp

Description This command applies a tunnel table export policy to LDP for the purpose of learning BGP labeled routes from the CPM tunnel table and stitching them to LDP FEC for the same prefix.

The user enables the stitching of routes between LDP and BGP by configuring separately tunnel table route export policies in both protocols and enabling the advertising of RFC 3107, *Carrying Label Information in BGP-4*, formatted labeled routes for prefixes learned from LDP FECs.

The route export policy in BGP instructs BGP to listen to LDP route entries in the CPM Tunnel Table. If a /32 LDP FEC prefix matches an entry in the export policy, BGP originates a BGP labeled route, stitches it to the LDP FEC, and re-distributes the BGP labeled route to its iBGP neighbors.

The user adds LDP FEC prefixes with the statement '**from protocol ldp**' in the configuration of the existing BGP export policy at the global level, the peer-group level, or at the peer level using the commands:

- **configure>router>bgp>export** *policy-name*
- **configure>router>bgp>group>export** *policy-name*
- **configure>router>bgp>group>neighbour>export** *policy-name*

To indicate to BGP to evaluate the entries with the **from protocol ldp** statement in the export policy when applied to a specific BGP neighbor, a new argument is added to the existing advertise-label command:

configure>router>bgp>group>neighbour>advertise-label ipv4 include-ldp-prefix

Without the **include-ldp-prefix** argument, only core IPv4 routes learned from RTM are advertised as BGP labeled routes to the neighbor. No stitching of LDP FEC to the BGP labeled route will be performed for this neighbor even if the same prefix was learned from LDP.

The tunnel table route export policy in LDP instructs LDP to listen to BGP route entries in the CPM Tunnel Table. If a /32 BGP labeled route matches a prefix entry in the export policy, LDP originates an LDP FEC for the prefix, stitches it to the BGP labeled route, and re-distributes the LDP FEC to its iBGP neighbors.

The user can add BGP labeled route prefixes with the statement '**from protocol bgp**' in the configuration of the LDP tunnel table export policy. Note that the '**from protocol**' statement has an effect only when the protocol value is ldp. Policy entries with protocol values of rsvp, bgp, or any value other than ldp are ignored at the time the policy is applied to LDP.

The **no** form of the command removes the policy from the configuration.

Default **no export-tunnel-table** — no tunnel table export route policy is specified.

Parameters *policy-name* — The export-tunnel-table route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters excluding double quotes. If the string contains spaces, use double quotes to delimit the start and end of the string. The specified name(s) must already be defined.

fec-originate

Syntax **fec-originate** *ip-prefix/mask* [**advertised-label** *in-label*] [**swap-label** *out-label*] **interface** *interface-name*
fec-originate *ip-prefix/mask* [**advertised-label** *in-label*] **next-hop** *ip-address* [**swap-label** *out-label*]
fec-originate *ip-prefix/mask* [**advertised-label** *in-label*] **next-hop** *ip-address* [**swap-label** *out-label*] **interface** *interface-name*
fec-originate *ip-prefix/mask* [**advertised-label** *in-label*] **pop**
no fec-originate *ip-prefix/mask* **interface** *interface-name*
no fec-originate *ip-prefix/mask* **next-hop** *ip-address*
no fec-originate *ip-prefix/mask* **next-hop** *ip-address* **interface** *interface-name*
no fec-originate *ip-prefix/mask* **pop**

Context config>router>ldp

Description This command defines a way to originate a FEC (with a swap action) for which the LSR is not egress, or to originate a FEC (with a pop action) for which the LSR is egress.

Parameters *ip-prefix/mask* — Specify information for the specified IP prefix and mask length.

Values <ip-address/mask> ipv4-prefix - a.b.c.d
ipv4-prefix-le - [0..32]
ipv6-prefix x:x:x:x:x:x:x:x (eight 16-bit pieces)
x:x:x:x:x:x:d.d.d.d
x - [0..FFFF]H
d - [0..255]D
ipv6-prefix-le - [0..128]

next-hop — Specify the IP address of the next hop of the prefix.

advertised-label — Specify the label advertised to the upstream peer. If not configured, then the label advertised should be from the label pool. If the configured static label is not available then the IP prefix is not advertised.

out-label — Specify the LSR to swap the label. If configured, then the LSR should swap the label with the configured swap-label. If not configured, then the default action is pop if the next-hop parameter is not defined.

NOTE: The next-hop, advertised-label, swap-label parameters are all optional. If next-hop is configured but no swap label specified, then it will be a swap with label 3, such as, pop and forward to the next-hop. If the next-hop and swap-label are configured, then it is a regular swap. If no parameters are specified, then a pop and route is performed.

Values 16 — 1048575

in-label — Specifies the number of labels to send to the peer associated with this FEC.

Values 32 — 1023

pop — Specifies to pop the label and transmit without the label.

interface *interface-name* **—** Specifies the name of the interface the label for the originated FEC is swapped to. For an unnumbered interface, this parameter is mandatory since there is no address for the next-hop. For a numbered interface, it is optional.

graceful-restart

Syntax [**no**] **graceful-restart**

Context config>router>ldp

Description This command enables graceful restart helper.

The **no** form of the command disables graceful restart. Note that graceful restart helper configuration changes, enable/disable or change of a parameter, will cause the LDP session to bounce.

Default **no graceful-restart** (**disabled**) — Graceful-restart must be explicitly enabled.

implicit-null-label

Syntax [**no**] **implicit-null-label**

Context config>router>ldp

Description This command enables the use of the implicit null label. Use this command to signal the IMPLICIT NULL option for all LDP FECs for which this node is the egress LER.

The **no** form of this command disables the signaling of the implicit null label.

Default **no implicit-null-label**

maximum-recovery-time

Syntax **maximum-recovery-time** *interval*
no maximum-recovery-time

Context config>router>ldp>graceful-restart

Description This command configures the local maximum recovery time.

The **no** form of the command returns the default value.

Default 120

Parameters *interval* — Specifies the length of time in seconds.

Values 15 — 1800

neighbor-liveness-time

Syntax **neighbor-liveness-time** *interval*
no neighbor-liveness-time

Context config>router>ldp>graceful-restart

Description This command configures the neighbor liveness time.

The **no** form of the command returns the default value.

Default 120

Parameters *interval* — Specifies the length of time in seconds.

Values 5 — 300

import

Syntax **import** *policy-name* [*policy-name …*up to 5 max]
no import

Context config>router>ldp

Description This command configures import route policies to determine which label bindings (FECs) are accepted from LDP neighbors. Policies are configured in the **config>router>policy-options** context.

If no import policy is specified, LDP accepts all label bindings from configured LDP neighbors. Import policies can be used to limit or modify the routes accepted and their corresponding parameters and metrics.

If multiple policy names are specified, the policies are evaluated in the order they are specified. The first policy that matches is applied. If multiple import commands are issued, the last command entered will override the previous command. A maximum of five policy names can be specified.

The **no** form of the command removes all policies from the configuration.

Default **no import** — No import route policies specified.

Parameters *policy-name* — The import route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

The specified name(s) must already be defined.

label-withdrawal-delay

Syntax **label-withdrawal-delay** *seconds*
no label-withdrawal-delay

Context config>router>ldp

Description This command specifies configures the time interval, in seconds, LDP will delay for the withdrawal of FEC-label binding it distributed to its neighbors when FEC is de-activated. When the timer expires, LDP then sends a label withdrawal for the FEC to all its neighbors. This is applicable only to LDP IPv4 prefix FECs and is not applicable to pseudowires (service FECs).

When there is an upper layer (user of LDP) which depends of LDP control plane for failover detection then label withdrawal delay and tunnel-down-damp-time options must be set to 0.

An example is PW redundancy where the primary PW doesn't have its own fast failover detection mechanism and the node depends on LDP tunnel down event to activate the standby PW.

Default no label-withdrawal-delay

Parameters *seconds* — Specifies the time that LDP delays the withdrawal of FEC-label binding it distributed to its neighbors when FEC is de-activated.

Values 3 — 120

mcast-upstream-frr

Syntax [**no**] **mcast-upstream-frr**

Context config>router>ldp

Description When LDP programs the primary ILM record in the data path, it provides the IOM with the This command enables the mLDP fast upstream switchover feature.

When this command is enabled and LDP is resolving a mLDP FEC received from a downstream LSR, it checks if an ECMP next-hop or a LFA next-hop exist to the root LSR node. If LDP finds one, it programs a primary ILM on the interface corresponding to the primary next-hop and a backup ILM on the interface corresponding to the ECMP or LFA next-hop. LDP then sends the corresponding labels to both upstream LSR nodes. In normal operation, the primary ILM accepts packets while the backup ILM drops them. If the interface or the upstream LSR of the primary ILM goes down causing the LDP session to go down, the backup ILM will then start accepting packets.

In order to make use of the ECMP next-hop, the user must configure the **ecmp** value in the system to at least 2 using the following command:

configure>router>ecmp

In order to make use of the LFA next-hop, the user must enable LFA using the following commands:

config>router>isis>loopfree-alternate

config>router>ospf>loopfree-alternate

Enabling IP FRR or LDP FRR features is not strictly required since LDP only needs to know where the alternate next-hop to the root LSR is to be able to send the Label Mapping message to program the backup ILM at the initial signaling of the tree. Thus enabling the LFA option is sufficient. If however, unicast IP and LDP prefixes need to be protected, then these features and the mLDP fast upstream switchover can be enabled concurrently.

Note that mLDP FRR fast switchover relies on the fast detection of loss of **LDP session** to the upstream peer to which primary ILM label had been advertised. As a result it is strongly recommended to perform the following:

- Enable BFD on all LDP interfaces to upstream LSR nodes. When BFD detects the loss of the last adjacency to the upstream LSR, it will bring down immediately the LDP session which will cause the IOM to activate the backup ILM.
- If there is a concurrent TLDP adjacency to the same upstream LSR node, enable BFD on the T-LDP peer in addition to enabling it on the interface.
- Enable the **ldp-sync-timer** option on all interfaces to the upstream LSR nodes. If an LDP session to the upstream LSR to which the primary ILM is resolved goes down for any other reason than a failure of the interface or of the upstream LSR, routing and LDP will go out of sync. This means the backup ILM will remain activated until the next time SPF is rerun by IGP. By enabling IGP-LDP synchronization feature, the advertised link metric will be changed to max value as soon as the LDP session goes down. This in turn will trigger an SPF and LDP will likely download a new set of primary and backup ILMs.

The **no** form of this command disables the fast upstream switchover for mLDP FECs.

Default no mcast-upstream-frr

tunnel-down-damp-time

Syntax **tunnel-down-damp-time** *seconds*
no tunnel-down-damp-time

Context config>router>ldp

Description This command specifies the time interval, in seconds, that LDP waits before posting a tunnel down event to the Tunnel Table Manager (TTM).

When LDP can no longer resolve a FEC and de-activates it, it de-programs the NHLFE in the data path. It will however delay deleting the LDP tunnel entry in the TTM until the tunnel-down-damp-time timer expires. This means users of the LDP tunnel, such as SDPs (all services) and BGP (L3 VPN), will not be notified immediately. Traffic is still blackholed because the forwarding engine NHLFE has been de-programmed.

If the FEC gets resolved before the tunnel-down-damp-time timer expires, then LDP programs the forwarding engine with the new NHLFE and performs a tunnel modify event in TTM updating the dampened entry in TTM with the new NHLFE information. If the FEC does not get resolved and the tunnel-down-damp-time timer expires, LDP posts a tunnel down event to TTM which deletes the LDP tunnel.

When there is an upper layer (user of LDP) which depends of LDP control plane for failover detection then label withdrawal delay and tunnel-down-damp-time options must be set to 0.

An example is pseudowire redundancy where the primary PW doesn't have its own fast failover detection mechanism and the node depends on LDP tunnel down event to activate the standby PW.

The **no** form of this command then tunnel down events are not damped.

Parameters *seconds* — Specifies the time interval, in seconds, that LDP waits before posting a tunnel down event to the Tunnel Table Manager.

keepalive

Syntax **keepalive** *timeout factor*
no keepalive

Context config>router>ldp>if-params>interface>ipv4
config>router>ldp>if-params>interface>ipv6
config>router>ldp>if-params>ipv4
config>router>ldp>if-params>ipv6
config>router>ldp>targeted-session>ipv4
config>router>ldp>targeted-session>ipv6
config>router>ldp>targ-session>peer
config>router>ldp>targ-session>peer-template

Description This command configures the time interval, in seconds, that LDP waits before tearing down the session. The **factor** parameter derives the keepalive interval.

If no LDP messages are exchanged for the configured time interval, the LDP session is torn down. Keepalive timeout is usually three times the keepalive interval. To maintain the session permanently, regardless of the activity, set the value to zero.

When LDP session is being set up, the keepalive timeout is negotiated to the lower of the two peers. Once a operational value is agreed upon, the keepalive factor is used to derive the value of the keepalive interval.

The **no** form of the command at the interface-parameters and targeted-session levels sets the **keepalive timeout** and the **keepalive factor** to the default value.

The **no** form of the command, at the interface level, sets the **keepalive timeout** and the **keepalive factor** to the value defined under the **interface-parameters** level.

The **no** form of the command, at the peer level, will set the **keepalive timeout** and the **keepalive factor** to the value defined under the **targeted-session** level.

Note that the session needs to be flapped for the new args to operate.

Default

Context	timeout	factor
config>router>ldp>if-params	30	3
config>router>ldp>targ-session	40	4
config>router>ldp>if-params>if	Inherits values from interface-parameters context.	
config>router>ldp>targ-session>peer	Inherits values from targeted-session context.	

Parameters *timeout* — Configures the time interval, expressed in seconds, that LDP waits before tearing down the session.

Values 1 — 65535

factor — Specifies the number of keepalive messages, expressed as a decimal integer, that should be sent on an idle LDP session in the keepalive timeout interval.

Values 1 — 255

local-lsr-id

Syntax **local-lsr-id** {**system** | **interface** | **interface-name** *interface-name*}
no local-lsr-id

Context config>router>ldp>interface-parameters>interface>ipv4/
config>router>ldp>interface-parameters>interface>ipv6
config>router>ldp>targeted-session>peer
config>router>ldp>targeted-session>peer-template

Description This command enables the use of the address of the local LDP interface, or any other network interface configured on the system, as the LSR-ID to establish link LDP Hello adjacency and LDP session with directly connected LDP peers. The network interface can be a loopback or not.

Link LDP sessions to all peers discovered over a given LDP interface share the same local LSR-ID. However, LDP sessions on different LDP interfaces can use different network interface addresses as their local LSR-ID.

By default, the LDP session to a peer uses the system interface address as the LSR-ID unless explicitly configured using the above command. Note, however, that the system interface must always be configured on the router, or the LDP protocol will not come up on the node. There is no requirement to include it in any routing protocol.

At initial configuration, the LDP session to a peer will remain down while the network interface used as LSR-ID is down. LDP will not try to bring it up using the system interface.

At any time the network IP interface used as LSR-ID goes down, the LDP sessions to all discovered peers using this LSR-ID go down.

If the user changes the LSR-ID value on the fly between **system**, **interface**, and *interface-name* while the LDP session is up, LDP will immediately tear down all sessions using this LSR-ID and will attempt to re-establish them using the new LSR-ID.

Note that when an interface other than system is used as the LSR-ID, the transport connection (TCP) for the link LDP session will also use the address of that interface as the transport address. If **system** or **interface** value is configured in the **configure>router>ldp>interface-parameters>interface>ipv4/ipv6>transport-address** context, it will be overridden.

The **no** form of the command returns to the default behavior in which case the system interface address is used as the LSR-ID.

Default no local-lsr-id

Parameters *interface-name* — Specifies the name, up to 32 character in length, of the network IP interface. AN interface name cannot be in the form of an IP address. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

tunneling

Syntax [**no**] **tunneling**

Context config>router>ldp>targ-session>peer
config>router>ldp>targ-session>peer-template

Description This command enables LDP over tunnels.

The **no** form of the command disables tunneling.

Default **no tunneling**

lsp

Syntax [**no**] **lsp** *lsp-name*

Context config>router>ldp>targ-session>tunneling

Description This command configures a specific LSP destined to this peer and to be used for tunneling of LDP FEC over RSVP. A maximum of 4 RSVP LSPs can be explicitly used for tunneling LDP FECs to the T-LDP peer.

It is not necessary to specify any RSVP LSP in this context unless there is a need to restrict the tunneling to selected LSPs. All RSVP LSPs with a to address matching that of the T-LDP peer are eligible by default. The user can also exclude specific LSP names by using the ldp-over-rsvp exclude command in the **configure->router->mpls->lsp** *lsp-name* context.

Interface Parameters Commands

interface-parameters

Syntax **interface-parameters**

Context config>router>ldp

Description This command enables the context to configure LDP interfaces and parameters applied to LDP interfaces. The user can configure different default parameters for IPv4 and IPv6 LDP interfaces by entering **ipv4** or **ipv6** as the next command.

ipv4

Syntax **ipv4**

Context config>router>ldp>interface parameters

This command enables the context to configure LDP interfaces and parameters applied to an IPv4 LDP interface.

ipv6

Syntax **ipv6**

Context config>router>ldp>interface parameters

Description This command enables the context to configure LDP interfaces and parameters applied to an IPv6 LDP interface.

hello

Syntax **hello** *timeout factor*
no hello

Context config>router>ldp>if-params>interface>ipv4
config>router>ldp>if-params>interface>ipv6
config>router>ldp>if-params>ipv4
config>router>ldp>if-params>ipv6
config>router>ldp>targ-session>ipv4
config>router>ldp>targ-session>ipv6
config>router>ldp>targ-session>peer
config>router>ldp>targ-session>peer-template

Description This command configures the time interval to wait before declaring a neighbor down. The **factor** parameter derives the hello interval.

Hold time is local to the system and sent in the hello messages to the neighbor. Hold time cannot be less than three times the hello interval. The hold time can be configured globally (applies to all LDP interfaces) or per interface. The most specific value is used.

When LDP session is being set up, the holddown time is negotiated to the lower of the two peers. Once a operational value is agreed upon, the hello factor is used to derive the value of the hello interval.

The **no** form of the command at the interface-parameters and targeted-session level sets the **hello timeout** and the **hello factor** to the default values.

The **no** form of the command, at the interface level, will set the **hello timeout** and the **hello factor** to the value defined under the interface-parameters level.

The **no** form of the command, at the peer level, will set the **hello timeout** and the **hello factor** to the value defined under the targeted-session level.

Note that the session needs to be flapped for the new args to operate.

Default

Context	Timeout	Factor
config>router>ldp>if-params	15	3
config>router>ldp>targ-session	45	3
config>router>ldp>if-params>if	Inherits values from interface-parameters context.	
config>router>ldp>targ-session>peer	Inherits values from targeted-session context.	

Parameters *timeout* — Configures the time interval, in seconds, that LDP waits before a neighbor down.

Values 1 — 65535

factor — Specifies the number of keepalive messages that should be sent on an idle LDP session in the hello timeout interval.

Values 1 — 255

hello-reduction

Syntax **hello-reduction** {**enable** *factor* | **disable**}
no hello-reduction

Context config>router>ldp>targeted-session>ipv4
config>router>ldp>targeted-session>ipv6
config>router>ldp>targeted-session>peer
config>router>ldp>targ-session>peer-template

Description This command enables the suppression of periodic targeted Hello messages between LDP peers once the targeted LDP session is brought up.

When this feature is enabled, the target Hello adjacency is brought up by advertising the Hold-Time value the user configured in the "**hello** timeout" parameter for the targeted session. The LSR node will then start advertising an exponentially increasing Hold-Time value in the Hello message as soon as the targeted LDP session to the peer is up. Each new incremented Hold-Time value is sent in a number of Hello messages equal to the value of the argument *factor*, which represents the dampening factor, before the next exponential value is advertised. This provides time for the two peers to settle on the new value. When the Hold-Time reaches the maximum value of 0xffff (binary 65535), the two peers will send Hello messages at a frequency of every [(65535-1)/local helloFactor] seconds for the lifetime of the targeted-LDP session (for example, if the local Hello Factor is three (3), then Hello messages will be sent every 21844 seconds.

The LSR node continues to compute the frequency of sending the Hello messages based on the minimum of its local Hold-time value and the one advertised by its peer as in RFC 5036. Thus for the targeted LDP session to suppress the periodic Hello messages, both peers must bring their advertised Hold-Time to the maximum value. If one of the LDP peers does not, the frequency of the Hello messages sent by both peers will continue to be governed by the smaller of the two Hold-Time values.

When the user enables the hello reduction option on the LSR node while the targeted LDP session to the peer is operationally up, the change will take effect immediately. In other words, the LSR node will start advertising an exponentially increasing Hold-Time value in the Hello message, starting with the current configured Hold-Time value.

When the user disables the hello reduction option while the targeted LDP session to the peer is operationally up, the change in the Hold-Time from 0xffff (binary 65535) to the user configured value for this peer will take effect immediately. The local LSR will immediately advertise the value of the user configured Hold-Time value and will not wait until the next scheduled time to send a Hello to make sure the peer adjusts its local hold timeout value immediately.

In general, any configuration change to the parameters of the T-LDP Hello adjacency (modifying the hello adjacency Hello Timeout or factor, enabling/disabling hello reduction, or modifying hello reduction factor) will cause the LSR node to trigger immediately an updated Hello message with the updated Hold Time value without waiting for the next scheduled time to send a Hello.

The **no** form of this command disables the hello reduction feature.

Default no hello-reduction

Parameters *factor* — Specifies the integer that specifies the Hello reduction dampening factor.

Values 3 —20

interface

Syntax [**no**] **interface** *ip-int-name* [**dual-stack**]

Context config>router>ldp>interface-parameters

Description This command enables LDP on the specified IP interface.

The **no** form of the command deletes the LDP interface and all configuration information associated with the LDP interface.

The LDP interface must be disabled using the **shutdown** command before it can be deleted.

The user can configure different parameters for IPv4 and IPv6 LDP interfaces by entering **ipv4** or **ipv6** as the next command.

Parameters *ip-int-name* — The name of an existing interface. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

dual-stack — This optional keyword allows the user to distinguish between configuration execs prior to SR OS Release 13.0 from those in Release 13.0, as the interface node implementation has changed in Release 13.0 to include new IPv4 and IPv6 contexts. The following are some of the key points for this keyword:

- If the keyword is provided, then IPv4 interface context will not be created. If it is not provided, the IPv4 interface context will be created. This will take care of execs of prior to Release 13.0 configurations on a router running SR OS Release 13.0.
- This new keyword will always show in a Release 13.0 configuration.
- When entering an already configured interface, there is no need to provide the keyword, but it will be ignored if provided.
- When deleting a configured interface, the keyword will not be accepted in the **no** version of the **interface** command.

ipv4

Syntax [**no**] **ipv4**

Context config>router>ldp>interface parameters>interface

This command enables the context to configure IPv4 LDP parameters applied to the interface.

ipv6

Syntax [**no**] **ipv6**

Context config>router>ldp>interface parameters>interface

Description This command enables the context to configure IPv6 LDP parameters applied to the interface.

bfd-enable

Syntax **bfd-enable** [**ipv4**][**ipv6**]
no bfd-enable

Context config>router>ldp>if-params>if

Description This command enables tracking of the Hello adjacency to an LDP peer using BFD.

When this command is enabled on an LDP interface, LDP registers with BFD and starts tracking the LSR-id of all peers it formed Hello adjacencies with over that LDP interface. The LDP hello mechanism is used to determine the remote address to be used for the BFD session. The parameters used for the BFD session, that is, transmit-interval, receive-interval, and multiplier are those configured under the IP interface in existing implementation: **config>router>interface>bfd**

The operation of BFD over an LDP interface tracks the next-hop of the IPv4 and IPv6 prefixes in addition to tracking the LDP peer address of the Hello adjacency over that link. This is required since LDP can resolve both IPv4 and IPv6 prefix FECs over a single IPv4 or IPv6 LDP session and as such the next-hop of a prefix will not necessarily match the LDP peer source address of the Hello adjacency.

The failure of either or both of the BFD session tracking the FEC next-hop and the one tracking the Hello adjacency will cause the LFA backup NHLFE for the FEC to be activated or the FEC to be re-resolved if there is no FRR backup.

When multiple links exist to the same LDP peer, a Hello adjacency is established over each link and a separate BFD session is enabled on each LDP interface. If a BFD session times out on a specific link, LDP will immediately associate the LDP session with one of the remaining Hello adjacencies and trigger the LDP FRR procedures. As soon as the last Hello adjacency goes down due to BFD timing out, the LDP session goes down and the LDP FRR procedures will be triggered.

The **no** form of this command disables BFD on the LDP interface.

Default no bfd-enable

transport-address

Syntax **transport-address** {**interface** | **system**}
no transport-address

Context config>router>ldp>interface-parameters>interface>ipv4
config>router>ldp>interface-parameters>interface>ipv6
config>router>ldp>interface-parameters>ipv4
config>router>ldp>interface-parameters>ipv6

Description This command configures the transport address to be used when setting up the LDP TCP sessions. The transport address can be configured as **interface** or **system**. The transport address can be configured globally (applies to all LDP interfaces) or per interface. The most specific value is used.

With the transport-address command, you can set up the LDP interface to the connection which can be set to the interface address or the system address. However, there can be an issue of which address to use when there are parallel adjacencies. This situation can not only happen with parallel links, it could be a link and a targeted adjacency since targeted adjacencies request the session to be set up only to the system IP address.

Note that the **transport-address** value should not be **interface** if multiple interfaces exist between two LDP neighbors. Depending on the first adjacency to be formed, the TCP endpoint is chosen. In other words, if one LDP interface is set up as **transport-address interface** and another for **transport-address system**, then, depending on which adjacency was set up first, the TCP endpoint addresses are determined. After that, because the hello contains the LSR ID, the LDP session can be checked to verify that it is set up and then match the adjacency to the session.

Note that for any given ILDP interface, as the **local-lsr-id** parameters is changed to **interface**, the **transport-address** configuration loses effectiveness. Since it will be ignored and the ILDP session will *always* use the relevant interface IP address as transport-address even though system is chosen.

The **no** form of the command, at the global level, sets the transport address to the default value. The **no** form of the command, at the interface level, sets the transport address to the value defined under the global level.

Default **system** — The system IP address is used.

Parameters **interface —** The IP interface address is used to set up the LDP session between neighbors. The transport address interface cannot be used if multiple interfaces exist between two neighbors, since only one LDP session is set up between two neighbors.

system — The system IP address is used to set up the LDP session between neighbors.

multicast-traffic

Syntax [**no**] **multicast-traffic**

Context config>router>ldp>interface-parameters>interface

Description This command enables P2MP multicast traffic forwarding on the interface.

The **no** form of command disables P2MP LDP multicast traffic on the interface. P2MP tree branching out on the interface would not withdraw label map from the peer session on interface shutdown or multicast traffic is disabled. Session may exist on multiple parallel interfaces. Only forwarding entry is changed when interface is shutdown or multicast traffic support is disabled.

Note that LDP may choose to egress the mLDP tree over this interface, but if multicast-traff is disabled, the dataplane will not forward traffic on this branch.

Default multicast-traffic enable

mp-mbb-time

Syntax [**no**] **mp-mbb-time**

Context config>router>ldp

Description This command configures the maximum time a P2MP transit/bud node must wait before switching over to the new path if the new node does not send MBB TLV to inform of the availability of data plane.

The **no** form of command should configure the default timer of 3 seconds.

Default 3 seconds

Parameters *interval* — seconds.

Values 1 — 10 seconds

Session Parameters Commands

session-parameters

Syntax **session-parameters**

Context config>router>ldp

Description This command enables the context to configure peer specific parameters.

peer

Syntax [**no**] **peer** *ip-address*

Context config>router>ldp>session-parameters

Description This command configures parameters for an LDP peer.

Default **none**

Parameters *ip-addr* — The IPv4 or IPv6 address of the LDP peer in dotted decimal notation.

adv-adj-addr-only

Syntax [**no**] **adv-adj-addr-only**

Context config>router>ldp>session-parameters>peer

Description This command provides a means for an LDP router to advertise only the local IPv4 or IPv6 interfaces it uses to establish hello adjacencies with an LDP peer. By default, when a router establishes an LDP session with a peer, it advertises in an LDP Address message the addresses of all local interfaces to allow the peer to resolve LDP FECs distributed by this router. Similarly, a router sends a Withdraw Address message to of all its peers to withdraw a local address if the corresponding interface went down or was deleted.

This new option reduces CPU processing when a large number of LDP neighbors come up or go down. The new CLI option is strongly recommended in mobile backhaul networks where the number of LDP peers can be very large.

The **no** form of this command reverts LDP to the default behavior of advertising all local interfaces.

dod-label-distribution

Syntax [**no**] **dod-label-distribution**

Context config>router>ldp>session-parameters>peer

Description This command enables the use of the LDP Downstream-on-Demand (DoD) label distribution procedures.

When this option is enabled, LDP will set the A-bit in the Label Initialization message when the LDP session to the peer is established. When both peers set the A-bit, they will both use the DoD label distribution method over the LDP session (RFC 5036).

This feature can only be enabled on a link-level LDP session and therefore will apply to prefix labels only, not service labels.

As soon as the link LDP session comes up, the 7x50 will send a label request to its DoD peer for the FEC prefix corresponding to the peer's LSR-id. The DoD peer LSR-id is found in the basic Hello discovery messages the peer used to establish the Hello adjacency with the 7x50.

Similarly if the 7x50 and the directly attached DoD peer entered into extended discovery and established a targeted LDP session, the 7x50 will immediately send a label request for the FEC prefix corresponding to the peer's LSR-id found in the extended discovery messages.

However, the 7x50 node will not advertise any <FEC, label> bindings, including the FEC of its own LSR-id, unless the DoD peer requested it using a Label Request Message.

When the DoD peer sends a label request for any FEC prefix, the 7x50 will reply with a <FEC, label> binding for that prefix if the FEC was already activated on the 7x50. If not, the 7x50 replies with a notification message containing the status code of "no route." The 7x50 will not attempt in the latter case to send a label request to the next-hop for the FEC prefix when the LDP session to this next-hop uses the DoD label distribution mode. Hence the reference to single-hop LDP DoD procedures.

As soon as the link LDP session comes up, the 7x50 will send a label request to its DoD peer for the FEC prefix corresponding to the peer's LSR-id. The DoD peer LSR-id is found in the basic Hello discovery messages the peer used to establish the Hello adjacency with the 7x50.

Similarly if the 7x50 and the directly attached DoD peer entered into extended discovery and established a targeted LDP session, the 7x50 will immediately send a label request for the FEC prefix corresponding to the peer's LSR-id found in the extended discovery messages. Peer address has to be the peer LSR-ID address.

The **no** form of this command disables the DoD label distribution with an LDP neighbor.

Default no dod-label-distribution

export-addresses

Syntax **export-addresses** *policy-name* [*policy-name*...(up to 5 max)]
no export-addresses

Context config>router>ldp>session-parameters>peer

Description This command specifies the export prefix policy to local addresses advertised to this peer.

Policies are configured in the **config>router>policy-options** context. A maximum of five policy names can be specified.

The **no** form of the command removes the policy from the configuration.

Parameters *policy-name* — The export-prefix route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters excluding double quotes. If the string contains spaces, use double quotes to delimit the start and end of the string. The specified name(s) must already be defined.

export-prefixes

Syntax [**no**] **export-prefixes** *policy-name*

Context config>router>ldp>session-parameters>peer

Description This command specifies the export route policy used to determine which prefixes received from other LDP and T-LDP peers are re-distributed to this LDP peer via the LDP/T-LDP session to this peer. A prefix that is filtered out (deny) will not be exported. A prefix that is filtered in (accept) will be exported.

If no export policy is specified, all FEC prefixes learned will be exported to this LDP peer. This policy is applied in addition to the global LDP policy and targeted session policy.

Policies are configured in the **config>router>policy-options** context. A maximum of five policy names can be specified. Peer address has to be the peer LSR-ID address.

The **no** form of the command removes the policy from the configuration.

Default no export-prefixes - no export route policy is specified

Parameters *policy-name* — The export-prefix route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters excluding double quotes. If the string contains spaces, use double quotes to delimit the start and end of the string. The specified name(s) must already be defined.

fec-limit

Syntax **fec-limit** *limit* [**log-only**] [**threshold** *percentage*]
no fec-limit

Context config>router>ldp>session-parameters>peer

Description This command configures a limit on the number of FECs which an LSR will accept from a given peer and add into the LDP label database. The limit applies to the aggregate count of all FEC types including service FEC. Once the limit is reached, any FEC received will be released back to the peer. This behavior is different from the per-peer import policy which will still accept the FEC into the label database but will not resolve it.

When the FEC limit for a peer is reached, the LSR performs the following actions:

1. Generates a trap and a syslog message.
2. Generates a LDP notification message with the LSR overload status TLV, for each LDP FEC type including service FEC, to this peer only if this peer advertised support for the LSR overload sub-TLV via the LSR Overload Protection Capability TLV at session initialization.
3. Releases, with LDP Status Code of "No_Label_Resources", any new FEC, including service FEC, from this peer which exceeds the limit.

Note that if a legitimate FEC is released back to a peer, while the FEC limit was exceeded, the user must have a means to replay that FEC back to the 7x50 LSR once the condition clears. This is done automatically if the peer is a 7x50 and supports the LDP overload status TLV (SROS 11.0R5 and higher). Third party peer implementations would need to support the LDP overload status TLV or provide a manual command to replay the FEC.

The **threshold percent** option allows to set a threshold value when a trap and an syslog message are generated as a warning to the user in addition to when the limit is reached. The default value for the threshold when not configured is 90%.

The **log-only** option causes a trap and syslog message to be generated when reaching the threshold and limit. However, LDP labels are not released back to the peer.

If the user decreases the limit value such that it is lower than the current number of FECs accepted from the peer, the LDP LSR raises the trap for exceeding the limit. In addition, it will set overload for peers which signaled support for LDP overload protection capability TLV. However, no existing resolved FECs from the peer which does not support the overload protection capability TLV should be de-programmed or released.

A different trap is released when crossing the threshold in the upward direction, when reaching the FEC limit, and when crossing the threshold in the downward direction. However the same trap will not be generated more often than 2 minutes apart if the number of FECs oscillates around the threshold or the FEC limit.

Default no fec-limit

Parameters *limit* — Specify the aggregate count of FECs of all types which can be accepted from this LDP peer.

log-only — Specify only a trap and syslog message are generated when reaching the threshold and limit. However, LDP labels are not released back to the peer.

threshold percent — Specify the threshold value (as a percentage) that triggers a warning syslog message and trap to be sent.

fec129-cisco-interop

Syntax [**no**] **fec129-cisco-interop**

Context config>router>ldp>session-parameters>peer

Description This command specifies whether LDP will provide translation between non-compliant FEC 129 formats of Cisco. Peer LDP sessions must be manually configured towards the non-compliant Cisco PEs.

When enabled, Cisco non-compliant format will be used to send and interpret received label release messages i.e. the FEC129 SAII and TAII fields will be reversed.

When the disabled, Cisco non-compliant format will not be used or supported. Peer address has to be the peer LSR-ID address.

The **no** form of the command returns the default.

Default no fec129-cisco-interop

fec-type-capability

Syntax **fec-type-capability**

Context config>router>ldp>session-parameters>peer
config>router>ldp>interface-parameters>interface>ipv4
config>router>ldp>interface-parameters>interface>ipv6

Description This command enables or disables the advertisement of a FEC type on a given LDP session or Hello adjacency to a peer.

p2mp

Syntax **p2mp** {**enable** | **disable**}

Context config>router>ldp>session-parameters>peer>fec-type-capability

Description This command enables or disables P2MP FEC capability for the session.

p2mp-ipv4

Syntax **p2mp** {**enable** | **disable**}

Context config>router>ldp>interface-parameters>interface>>ipv4>fec-type-capability
config>router>ldp>interface-parameters>interface>>ipv6>fec-type-capability
config>router>ldp>session-parameters>peer>fec-type-capability

Description This command enables or disables IPv4 P2MP FEC capability on the interface.

p2mp-ipv6

Syntax **p2mp** {**enable** | **disable**}

Context config>router>ldp>interface-parameters>interface>>ipv4>fec-type-capability
config>router>ldp>interface-parameters>interface>>ipv6>fec-type-capability
config>router>ldp>session-parameters>peer>fec-type-capability

Description This command enables or disables IPv6 P2MP FEC capability on the interface.

prefix-ipv4

Syntax **prefix-ipv4** {**enable** | **disable**}

Context config>router>ldp>interface-parameters>interface>ipv4>fec-type-capability
config>router>ldp>interface-parameters>interface>ipv6>fec-type-capability
config>router>ldp>session-parameters>peer>fec-type-capability

Description This command enables or disables IPv4 prefix FEC capability on the session or interface.

prefix-ipv6

Syntax **prefix-ipv6** {**enable** | **disable**}

Context config>router>ldp>interface-parameters>interface>ipv4>fec-type-capability
config>router>ldp>interface-parameters>interface>ipv6>fec-type-capability
config>router>ldp>session-parameters>peer>fec-type-capability

Description This command enables or disables IPv6 prefix FEC capability on the session or interface.

import-prefixes

Syntax [**no**] **import-prefixes** *policy-name*

Context config>router>ldp>session-parameters>peer

Description This command configures the import FEC prefix policy to determine which prefixes received from this LDP peer are imported and installed by LDP on this node. If resolved these FEC prefixes are then re-distributed to other LDP and T-LDP peers. A FEC prefix that is filtered out (deny) will not be imported. A FEC prefix that is filtered in (accept) will be imported.

If no import policy is specified, the node will import all prefixes received from this LDP/T-LDP peer. This policy is applied in addition to the global LDP policy and targeted session policy.

Policies are configured in the **config>router>policy-options** context. A maximum of five policy names can be specified. Peer address has to be the peer LSR-ID address.

The **no** form of the command removes the policy from the configuration.

Default no import-prefixes - no import route policy is specified

Parameters *policy-name* — The import-prefix route policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters excluding double quotes. If the string contains spaces, use double quotes to delimit the start and end of the string. The specified name(s) must already be defined.

path-mtu-discovery

Syntax **path-mtu-discovery**
no path-mtu-discovery

Context config>router>ldp>tcp-session-parameters>peer-transport

Description This command enables Path MTU discovery for the associated TCP connections. When enabled, the MTU for the associated TCP session is initially set to the egress interface MTU. The DF bit is also set so that if a router along the path of the TCP connection cannot handle a packet of a particular size without fragmenting, it sends back and ICMP message to set the path MTU for the given session to a lower value that can be forwarded without fragmenting. Note that if one or more transport addresses used in the Hello adjacencies to the same peer LSR are different from the LSR-ID value, the user must add each of the transport addresses to the path MTU discovery configuration as a separate peer. This means when the TCP connection is bootstrapped by a given Hello adjacency, the path MTU discovery can operate over that specific TCP connection by using its specific transport address.

Default **no path-mtu-discovery**

pe-id-mac-flush-interop

Syntax [**no**] **pe-id-mac-flush-interop**

Context config>router>ldp>session-parameters>peer

Description This command enables the addition of the PE-ID TLV in the LDP MAC withdrawal (mac-flush) message, under certain conditions, and modifies the mac-flush behavior for interoperability with other vendors that do not support the flush-all-from-me vendor-specific TLV. This flag can be enabled on a per LDP peer basis and allows the flush-all-from-me interoperability with other vendors. When the pe-id-mac-flush-interop flag is enabled for a given peer, the current mac-flush behavior is modified in terms of mac-flush generation, mac-flush propagation and behavior upon receiving a mac-flush.

The mac-flush generation will be changed depending on the type of event and according to the following rules:

- Any all-from-me mac-flush event will trigger a mac-flush all-but-mine message (RFC 4762 compliant format) with the addition of a PE-ID TLV. The PE-ID TLV contains the IP address of the sending PE.
- Any all-but-mine mac-flush event will trigger a mac-flush all-but-mine message WITHOUT the addition of the PE-ID TLV, as long as the source spoke-sdp is not part of an end-point.
- Any all-but-mine mac-flush event will trigger a mac-flush all-but-mine message WITH the addition of the PE-ID TLV, if the source spoke-sdp is part of an end-point and the spoke-sdp goes from down/standby state to active state. In this case, the PE-ID TLV will contain the IP address of the PE to which the previous active spoke-sdp was connected to.

Any other case will follow the existing mac-flush procedures.

When the pe-id-mac-flush-interop flag is enabled for a given LDP peer, the mac-flush ingress processing is modified according to the following rules:

- Any received all-from-me mac-flush will follow the existing mac-flush all-from-me rules regardless of the existence of the PE-ID.
- Any received all-but-mine mac-flush will take into account the received PE-ID, i.e. all the mac addresses associated to the PE-ID will be flushed. If the PE-ID is not included, the mac addresses associated to the sending PE will be flushed.
- Any other case will follow the existing mac-flush procedures.

When a mac-flush message has to be propagated (for an ingress sdp-binding to an egress sdp-binding) and the pe-id-mac-flush-interop flag is enabled for the ingress and egress TLDP peers, the following behavior is observed:

- If the ingress and egress bindings are spoke-sdp, the PE will propagate the mac-flush message with its own PE-ID.

- If the ingress binding is an spoke-sdp and the egress binding a mesh-sdp, the PE will propagate the mac-flush message without modifying the PE-ID included in the PE-ID TLV.
- If the ingress binding is a mesh-sdp and the egress binding an spoke-sdp, the PE will propagate the mac-flush message with its own PE-ID.
- When ingress and egress bindings are mesh-sdp, the mac-flush message is never propagated. This is the behavior regardless of the pe-id-mac-flush-interop flag configuration.

Note that the PE-ID TLV is never added when generating a mac-flush message on a B-VPLS if the send-bvpls-flush command is enabled in the I-VPLS. In the same way, no PE-ID is added when propagating mac-flush from a B-VPLS to a I-VPLS when the propagate-mac-flush-from-bvpls command is enabled. Mac-flush messages for peers within the same I-VPLS or within the same B-VPLS domain follow the procedures described above.

Default no pe-id-mac-flush-interop

prefer-tunnel-in-tunnel

Syntax [**no**] **prefer-tunnel-in-tunnel**

Context config>router>ldp

Description This command specifies to use tunnel-in-tunnel over a simple LDP tunnel. Specifically, the user packets for LDP FECs learned over this targeted LDP session can be sent inside an RSVP LSP which terminates on the same egress router as the destination of the targeted LDP session. The user can specify an explicit list of RSVP LSP tunnels under the Targeted LDP session or LDP will perform a lookup in the Tunnel Table Manager (TTM) for the best RSVP LSP. In the former case, only the specified LSPs will be considered to tunnel LDP user packets. In the latter case, all LSPs available to the TTM and which terminate on the same egress router as this target ed LDP session will be considered. In both cases, the metric specified under the LSP configuration is used to control this selection.

Note that the lookup in the TTM will prefer a LDP tunnel over an LDP-over-RSVP tunnel if both are available. Also note that the tunneling operates on the dataplane only. Control packets of this targeted LDP session are sent over the IGP path.

shortcut-transit-ttl-propagate

Syntax [**no**] **shortcut-transit-ttl-propagate**

Context config>router>ldp
config>router>mpls

Description This command configures the TTL handling of transit packets for all LSP shortcuts originating on this ingress LER. It applies to all LDP or RSVP LSPs that are used to resolve static routes, BGP routes, and IGP routes.

The user can enable or disable the propagation of the TTL from the header of an IP packet into the header of the resulting MPLS packet independently for local and transit packets forwarded over an LSP shortcut.

By default, the feature propagates the TTL from the header of transit IP packets into the label stack of the resulting MPLS packets forwarded over the LSP shortcut. This is referred to as Uniform mode.

When the **no** form of the command is enabled, TTL propagation is disabled on all transit IP packets received on any IES interface and destined to a route that is resolved to the LSP shortcut. In this case, a TTL of 255 is programmed onto the pushed label stack. This is referred to as Pipe mode.

Default shortcut-transit-ttl-propagate

shortcut-local-ttl-propagate

Syntax [**no**] **shortcut-local-ttl-propagate**

Context config>router>ldp
config>router>mpls

Description This command configures the TTL handling of locally generated packets for all LSP shortcuts originating on this ingress LER. It applies to all LDP or RSVP LSPs that are used to resolve static routes, BGP routes, and IGP routes.

The user can enable or disable the propagation of the TTL from the header of an IP packet into the header of the resulting MPLS packet independently for local and transit packets forwarded over an LSP shortcut.

Local IP packets include ICMP Ping, traceroute, and OAM packets, that are destined to a route that is resolved to the LSP shortcut. Transit IP packets are all IP packets received on any IES interface and destined to a route that is resolved to the LSP shortcut

By default, the feature propagates the TTL from the header of locally generated IP packets into the label stack of the resulting MPLS packets forwarded over the LSP shortcut. This is referred to as Uniform mode.

When the **no** form of the above command is enabled, TTL propagation is disabled on all locally generated IP packets, including ICMP Ping, traceroute, and OAM packets, that are destined to a route that is resolved to the LSP shortcut. In this case, a TTL of 255 is programmed onto the pushed label stack. This is referred to as Pipe mode.

Default shortcut-local-ttl-propagate

Targeted Session Commands

targeted-session

Syntax **targeted-session**

Context config>router>ldp

Description This command configures targeted LDP sessions. Targeted sessions are LDP sessions between non-directly connected peers. Hello messages are sent directly to the peer platform instead of to all the routers on this subnet multicast address. The user can configure different default parameters for IPv4 and IPv6 LDP targeted hello adjacencies.

The discovery messages for an indirect LDP session are addressed to the specified peer and not to the multicast address.

Default none

bfd-enable

Syntax [**no**] **bfd-enable**

Context config>router>ldp>targ-session>peer
config>router>ldp>targeted-session>peer-template

Description This command enables the bidirectional forwarding detection (BFD) session for the selected TLDP session. By enabling BFD for a selected targeted session, the state of that session is tied to the state of the underneath BFD session between the two nodes.

The parameters used for the BFD are set via the BFD command under the IP interface.

The **no** form of this command removes the TLDP session operational state binding to the central BFD session one.

Default no bfd-enable

disable-targeted-session

Syntax [**no**] **disable-targeted-session**

Context config>router>ldp>targ-session

Description This command disables support for SDP triggered automatic generated targeted sessions. Targeted sessions are LDP sessions between non-directly connected peers. The discovery messages for an indirect LDP session are addressed to the specified peer and not to the multicast address.

The **no** form of the command enables the set up of any targeted sessions.

Default **no disable-targeted-session**

peer

Syntax [**no**] **peer** *ip-address*

Context config>router>ldp>targeted-session

Description This command configures parameters for an LDP peer.no

Default **none**

Parameters *ip-address* — The IPv4 or IPv6 address of the LDP peer in dotted decimal notation.

peer-template-map

Syntax **peer-template-map** *template-name* **policy** *peer-prefix-policy1* [*peer-prefix-policy2..up to 5*]
no peer-template-map peer-template *template-name*

Context config>router>ldp>targeted-session

Description This command enables the automatic creation of a targeted Hello adjacency and LDP session to a discovered peer. The user configures a targeted session peer parameter template and binds it to a peer prefix policy.

Each application of a targeted session template to a given prefix in the prefix list will result in the establishment of a targeted Hello adjacency to an LDP peer using the template parameters as long as the prefix corresponds to a router-id for a node in the TE database. As a result of this, the user must enable the traffic-engineering option in ISIS or OSPF. The targeted Hello adjacency will either trigger a new LDP session or will be associated with an existing LDP session to that peer.

Up to 5 peer prefix policies can be associated with a single peer template at all times. Also, the user can associate multiple templates with the same or different peer prefix policies. Thus multiple templates can match with a given peer prefix. In all cases, the targeted session parameters applied to a given peer prefix are taken from the first created template by the user. This provides a more deterministic behavior regardless of the order in which the templates are associated with the prefix policies.

Each time the user executes the above command, with the same or different prefix policy associations, or the user changes a prefix policy associated with a targeted peer template, the system re-evaluates the prefix policy. The outcome of the re-evaluation will tell LDP if an existing targeted Hello adjacency needs to be torn down or if an existing targeted Hello adjacency needs to have its parameters updated on the fly.

If a /32 prefix is added to (removed from) or if a prefix range is expanded (shrunk) in a prefix list associated with a targeted peer template, the same prefix policy re-evaluation described above is performed.

The template comes up in the **no shutdown** state and as such it takes effect immediately. Once a template is in use, the user can change any of the parameters on the fly without shutting down the template. In this case, all targeted Hello adjacencies are updated.

The SR OS supports multiple ways of establishing a targeted Hello adjacency to a peer LSR:

- User configuration of the peer with the targeted session parameters inherited from the **config>router>ldp>targeted-session** in the top level context or explicitly configured for this peer in the **config>router>ldp>targeted-session>peer** context and which overrides the top level parameters shared by all targeted peers. Let us refer to the top level configuration context as the global context. Note that some parameters only exist in the global context and as such their value will always be inherited by all targeted peers regardless of which event triggered it.
- User configuration of an SDP of any type to a peer with the signaling tldp option enabled (default configuration). In this case the targeted session parameter values are taken from the global context.
- User configuration of a (FEC 129) PW template binding in a BGP-VPLS service. In this case the targeted session parameter values are taken from the global context.
- User configuration of a (FEC 129 type II) PW template binding in a VLL service (dynamic multi-segment PW). In this case the target session parameter values are taken from the global context
- User configuration of a mapping of a targeted session peer parameter template to a prefix policy when the peer address exists in the TE database (this feature). In this case, the targeted session parameter values are taken from the template.

Since the above triggering events can occur simultaneously or in any arbitrary order, the LDP code implements a priority handling mechanism in order to decide which event overrides the active targeted session parameters. The overriding trigger will become the owner of the targeted adjacency to a given peer. The following is the priority order:

- Priority 1: manual configuration of session parameters
- Priority 2: mapping of targeted session template to prefix policy.
- Priority 3: manual configuration of SDP, PW template binding in BGP-AD VPLS and in FEC 129 VLL.

Note that any parameter value change to an active targeted Hello adjacency caused by any of the above triggering events is performed on the fly by having LDP immediately send a Hello message

with the new parameters to the peer without waiting for the next scheduled time for the Hello message. This allows the peer to adjust its local state machine immediately and maintains both the Hello adjacency and the LDP session in UP state. The only exceptions are the following:

- The triggering event caused a change to the local-lsr-id parameter value. In this case, the Hello adjacency is brought down which will also cause the LDP session to be brought down if this is the last Hello adjacency associated with the session. A new Hello adjacency and LDP session will then get established to the peer using the new value of the local LSR ID.
- The triggering event caused the targeted peer shutdown option to be enabled. In this case, the Hello adjacency is brought down which will also cause the LDP session to be brought down if this is the last Hello adjacency associated with the session.

Finally, the value of any LDP parameter which is specific to the LDP/TCP session to a peer is inherited from the **config>router>ldp>session-parameters>peer** context. This includes MD5 authentication, LDP prefix per-peer policies, label distribution mode (DU or DOD), etc.

The no form of this command deletes the binding of the template to the peer prefix list and brings down all Hello adjacencies to the discovered LDP peers.

peer-template

Syntax [**no**] **peer-template template-name**

Context config>router>ldp>targeted-session

Description This command creates a targeted session peer parameter template that can be referenced in the automatic creation of targeted Hello adjacency and LDP session to a discovered peer.

The **no** form of command deletes the peer template. A peer template cannot be deleted if it is bound to a peer prefix list.

Parameters *template-name* — Specifies the template name to identify targeted peer template. It must be 32 characters maximum.

export-prefixes

Syntax **export-prefixes** *policy-name* [*policy-name*...(up to 5 max)]
no export-prefixes

Context config>router>ldp>targeted-session

Description This command specifies the export route policy used to determine which FEC prefix label bindings are exported from a targeted LDP session. A route that is filtered out (deny) will not be exported. A route that is filtered in (accept) will be exported.

If no export policy is specified, all bindings learned through a targeted LDP session will be exported to all targeted LDP peers. This policy is applied in addition to the global LDP policy.

Policies are configured in the **config>router>policy-options** context. A maximum of five policy names can be specified.

The **no** form of the command removes the policy from the configuration.

Parameters *policy-name* — The export policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

import-prefixes

Syntax **import-prefixes** *policy-name* [*policy-name*...(up to 5 max)]
no import-prefixes

Context config>router>ldp>targeted-session

Description This command configures the import route policy to determine which FEC prefix label bindings are accepted from targeted LDP neighbors into this node. A label binding that is filtered out (deny) will not be imported. A route that is filtered in (accept) will be imported.

If no import policy is specified, this node session will accept all bindings from configured targeted LDP neighbors. This policy is applied in addition to the global LDP policy.

Policies are configured in the **config>router>policy-option**s context. A maximum of five policy names can be specified.

The **no** form of the command removes the policy from the configuration.

Parameters *policy-name* — The import policy name. Allowed values are any string up to 32 characters long composed of printable, 7-bit ASCII characters. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

ipv4

Syntax **ipv4**

Context config>router>ldp>targeted-session

Description This command enables the context to configure parameters applied to targeted sessions to all IPv4 LDP peers.

ipv6

Syntax **ipv4**

Context config>router>ldp>targeted-session

Description This command enables the context to configure parameters applied to targeted sessions to all IPv6 LDP peers.

TCP Session Parameters Commands

tcp-session-parameters

Syntax **tcp-session-parameters**

Context config>router>ldp

Description This command enables the context to configure parameters applicable to TCP transport session of an LDP session to remote peer.

peer-transport

Syntax **peer-transport** *ip-address*
no peer transport

Context config>router>ldp>tcp-session-parameters

Description This command configures the peer transport address, that is, the destination address of the TCP connection, and not the address corresponding to the LDP LSR-ID of the peer.

Parameters *ip-address* — The IPv4 or IPv6 address of the TCP connection to the LDP peer in dotted decimal notation.

auth-keychain

Syntax **auth-keychain** *name*

Context config>router>ldp>tcp-session-parameters>peer-transport

Description This command configures TCP authentication keychain to use for the session.

Parameters *name* — Specifies the name of the keychain to use for the specified TCP session or sessions. This keychain allows the rollover of authentication keys during the lifetime of a session up to 32 characters in length. Peer address has to be the TCP session transport address.

authentication-key

Syntax **authentication-key** [*authentication-key* | *hash-key*] [**hash** | **hash2**]
no authentication-key

Context config>router>ldp>tcp-session-parameters>peer-transport

Description This command specifies the authentication key to be used between LDP peers before establishing sessions. Authentication uses the MD-5 message-based digest. Peer address has to be the TCP session transport address. Note that if one or more transport addresses used in the Hello adjacencies to the

same peer LSR are different from the LSR-ID value, the user must add each of the transport addresses to the authentication-key configuration as a separate peer. This means when the TCP connection is bootstrapped by a given Hello adjacency, the authentication can operate over that specific TCP connection by using its specific transport address.

The **no** form of this command disables authentication.

Default **none**

Parameters *authentication-key* — The authentication key. The key can be any combination of ASCII characters up to 16 characters in length (unencrypted). If spaces are used in the string, enclose the entire string in quotation marks (“ ”).

hash-key — The hash key. The key can be any combination of up 33 alphanumeric characters. If spaces are used in the string, enclose the entire string in quotation marks (“ ”).

This is useful when a user must configure the parameter, but, for security purposes, the actual unencrypted key value is not provided.

hash — Specifies the key is entered in an encrypted form. If the hash keyword is not used, the key is assumed to be in a non-encrypted, clear text form. For security, all keys are stored in encrypted form in the configuration file with the hash parameter specified.

hash2 — Specifies the key is entered in a more complex encrypted form. If the **hash2** parameter is not used, the less encrypted **hash** form is assumed.

path-mtu-discovery

Syntax **path-mtu-discovery**
no path-mtu-discovery

Context config>router>ldp>tcp-session-parameters>peer-transport

Description This command enables Path MTU discovery for the associated TCP connections. When enabled, the MTU for the associated TCP session is initially set to the egress interface MTU. The DF bit is also set so that if a router along the path of the TCP connection cannot handle a packet of a particular size without fragmenting, it sends back and ICMP message to set the path MTU for the given session to a lower value that can be forwarded without fragmenting.

Note that if one or more transport addresses used in the Hello adjacencies to the same peer LSR are different from the LSR-ID value, the user must add each of the transport addresses to the path MTU discovery configuration as a separate peer. This means when the TCP connection is bootstrapped by a given Hello adjacency, the path MTU discovery can operate over that specific TCP connection by using its specific transport address.

Default **no path-mtu-discovery**

ttl-security

Syntax **ttl-security** *min-ttl-value*
no ttl-security

Context config>router>ldp>tcp-session-parameters>peer-transport

Description This command configures TTL security parameters for incoming packets. When the feature is enabled, BGP/LDP will accept incoming IP packets from a peer only if the TTL value in the packet is greater than or equal to the minimum TTL value configured for that peer. Peer address has to be the TCP session transport address.

The **no** form of the command disables TTL security.

Default **no ttl-security**

Parameters *min-ttl-value* — Specify the minimum TTL value for an incoming packet.

Values 1 — 255