

Service-Layer Commands

This module outlines the commands necessary to utilize the service layer.

For detailed information about Service Layer concepts, configuration tasks, and examples, see the *Use Service Layer API to Bring your Controller on Cisco IOS XR Routerin the Cisco 8000 Series Router* module in the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

- show service-layer mpls label, on page 2
- show service-layer path-groups, on page 3
- show service-layer policy, on page 5
- show service-layer policy internal database path-groups, on page 7
- show service-layer rib notifications registrations, on page 10
- show service-layer route, on page 12

show service-layer mpls label

To display MPLS label information service-layer, use the **show service-layer mpls-label** command in EXEC mode.

show service-layer mpls label { *label-number* | brief | client { *client-id* | all | detail } }

Syntax Description

label-number	Specifies a particular label by its number in the range of 16 to 1048575.
brief	Shows brief information about MPLS labels.
client *client-id*	Displays information about a particular client in the range of 0 to 65535.
client all	Displays information about all clients.
detail	Shows more detailed information about MPLS labels.

Command Default

No specific guidelines impact the use of this command.

Command Modes

EXEC

Command History

Release	Modification
Release 24.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows output of **show service-layer mpls-label trace lib all** command.

```
Router#show service-layer mpls-label trace lib all

22 unique entries (512 possible, 0 filtered)
Oct 16 23:06:27.527 sl/mpls-label_debug 0/RP0/CPU0 1# t12931 ltrace init ok
Oct 16 23:06:27.537 sl/mpls-label_debug 0/RP0/CPU0 1# t12931 edm init ok
Oct 16 23:06:27.537 sl/mpls-label_debug 0/RP0/CPU0 1# t12931 request init ok
```

show service-layer path-groups

To display path group information in the service-layer, use the **show service-layer path-groups** command in EXEC mode.

show service-layer path-groups { brief | client { *client-id* **| all } | detail | name | vrf {** *vrf-name* **| all } }**

Syntax Description

brief	Shows brief information about path-groups.
client *client-id*	Shows information about a particular client ID in the range of 0 to 65535.
client all	Shows information about all clients.
detail	Shows more detailed information about the path groups.
name	Displays information about path-group name.
vrf *vrf-name*	Displays vrf table information for a specific vrf-name.
vrf all	Displays all the information for all vrfs.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 24.4.1	This command was introduced.
Release 26.1.1	The command output was modified to display the per-path load metric without the route-level metric.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows output of **show service-layer path-groups client all detail** command.

```
Router#show service-layer path groups client all detail


bsid_lspgrp2_eb1_eb2_gold_class-ipv6--1, tag: 0, distance: 30, Route flags(0xc): [viable
paths only, active on viable path]
    path: 1, fe80::ebb:bb03 (nexthop in vrf default) via Bundle-Ether1303
```

```
        remote labels: 62303,
        load metric: 0, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 401
  Client:50, Session:0, Operation ID:700000000001
  RIB:Programmed, FIB:Programmed, Ack Type:RIB
  Object Version:2
bsid_lspgrp2_eb1_eb2_gold_class-ipv6--10, tag: 0, distance: 30, Route flags(0xc): [viable
paths only, active on viable path]
    path: 1, fe80::ebb:bb03 (nexthop in vrf default) via Bundle-Ether1303
        remote labels: 62303,
        load metric: 0, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 401
  Client:50, Session:0, Operation ID:700000000021
  RIB:Programmed, FIB:Programmed, Ack Type:RIB
  Object Version:2
```

show service-layer policy

To display policy based routing information in the service-layer, use the **show service-layer policy** command in EXEC mode.

show service-layer policy { internal database path-groups | **rule-stats mappings { all** | **policy-name** *name-of-the-policy* **}** | **trace { lib { all** | **debug** | **error } }**

Syntax Description

internal	Displays internal policy-based routing information.
database	Displays policy-based routing database information.
path-groups	Displays path group information in the policy-based routing database.
rule-stats	Displays rule-stats mapping data.
mappings all	Retrieves global-key to local-key mappings.
mappings policy-name *name-of-the-policy*	Gets data for a specific policy name.
trace	Shows policy-based routing trace info.
lib	Shows service-layer policy library general traces.
all	Shows all general traces.
debug	Shows general debug traces.
error	Shows general error traces.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 24.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows output of **show service-layer policy trace lib all** command.

```
Router#show service-layer policy trace lib all

22 unique entries (512 possible, 0 filtered)
Oct 16 23:06:27.527 sl/policy_debug 0/RP0/CPU0 1# t12931 ltrace init ok
Oct 16 23:06:27.537 sl/policy_debug 0/RP0/CPU0 1# t12931 edm init ok
Oct 16 23:06:27.537 sl/policy_debug 0/RP0/CPU0 1# t12931 request init ok
```

show service-layer policy internal database path-groups

To display detailed information about path-groups in the service-layer policy database, use the **show service-layer policy internal database path-groups** command in EXEC mode.

show service-layer policy internal database path-groups

Syntax Description

internal	Displays internal policy-based routing information.
database	Displays policy-based routing database information.
path-groups	Displays path group information in the policy-based routing database.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 26.2.1	Introduced Path-Group Type column in the output.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Output syntax description

Syntax Description

Path-Group Str	Displays the name of the path-group.
PGRP ID	Displays the unique identifier assigned to the path-group.
Policy Name	Displays the name of the policy associated with the path-group.
Rule Name	Displays the rules within the policy that reference the path-group.
Path-Group Type	Indicates how the path-group is used in the rule within the policy: • Match: Used for matching traffic conditions • Action: Used for forwarding or policy actions • Match, Action: Used for both matching and action

INUSE State	Indicates whether the path-group usage state has been notified to RIB based on PBR programming in hardware. Possible values: • NONE: Policy not applied or not tracked • TRACKING: Awaiting hardware acknowledgment • INUSE-ACK: Successfully programmed in hardware • INUSE-NACK: Failed programming in hardware
INUSE State Timestamp	Timestamp of the most recent state update.

Example

This example shows output of **show service-layer policy trace lib all** command.

```
Router# show service-layer policy internal database path-groups
The output for INUSE columns should be interpreted as:

# INUSE State - Indicates whether the Path-Group INUSE state has been notified to RIB based
 on PBR programming in hardware.
  Possible values:
   NONE       - Either the Policy containing this Path-Group is not applied to an Interface
 (or any of its members), OR, this is a match Path-Group which is not tracked by HW-ACK
library.
   TRACKING   - This Path-Group is currently being tracked via the HW-ACK library, and is
pending responses from producers.
   INUSE-ACK  - This Path-Group was successfully programmed in PBR, and an ACK has been
notified to RIB.
   INUSE-NACK - This Path-Group failed PBR programming in hardware, and a NACK has been
notified to RIB.

# INUSE State Timestamp - Records the timestamp whenever the latest state of a Path-Group
is updated.

-------------------------------------------------------------------------------------------------
Path-Group Str   | PGRP ID    | Policy Name | Rule Name | Path-Group Type | INUSE State |
INUSE State Timestamp
-------------------------------------------------------------------------------------------------
path-group-4     | 0xa000004  | p1          | r1_p1     | Match           | NONE        |
Mar 17 2026 11:49:51.643
-------------------------------------------------------------------------------------------------
path-group-1     | 0xa000001  | p1          | r1_p1     | Action          | INUSE-ACK   |
Mar 17 2026 11:52:05.140
-------------------------------------------------------------------------------------------------
path-group-2     | 0xa000002  | p1          | r4_p1     | Match, Action   | INUSE-ACK   |
Mar 17 2026 11:52:05.137
                 |            |             | r2_p1     | Action          |             |
-------------------------------------------------------------------------------------------------
path-group-3     | 0xa000003  | p1          | r4_p1     | Match           | INUSE-ACK   |
Mar 17 2026 11:52:05.144
                 |            |             | r3_p1     | Action          |             |
-------------------------------------------------------------------------------------------------
path-group-6     | 0xa000006  | p1          | r3_p1     | Match           | NONE        |
Mar 17 2026 11:49:51.645
-------------------------------------------------------------------------------------------------
path-group-5     | 0xa000005  | p1          | r2_p1     | Match           | NONE        |
Mar 17 2026 11:49:51.644
```

```
--------------------------------------------------------------------------------------------------------------------
Router#
```

show service-layer rib notifications registrations

To display all incoming registration requests of route redistribution and nexthop tracking, use the **show service-layer rib notifications registrations** command in EXEC mode.

show service-layer route session *session-id*

Syntax Description

session	Specify the route redistribution and next hop tracking session.
session-id	Specifies the session ID.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 24.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

This example shows output of **show service-layer rib notifications registrations** command filtered by session ID.

```
Router#show service-layer rib notifications registrations session 1

IPv4 registrations:
  Session: 1, Client-id: 1, VRF: default
      Route Redistribution registrations:
          proto: local
      NextHop Tracking registrations:
          192.0.2.0/32
IPv6 registrations:
  Session: 1, Client-id: 1, VRF: default
      Route Redistribution registrations:
          proto: connected
      NextHop Tracking registrations:
          2001:0DB8:1:10B::/120, Allow-Default
```

This example shows output of **show service-layer rib notifications registrations** command filtered by session ID for Route redistribution registrations.

```
Router#show service-layer rib notifications registrations session 1 redistribution

IPv4 registrations:
  Session: 1, Client-id: 1, VRF: default
```

```
      Route Redistribution registrations:
          proto: local
IPv6 registrations:
  Session: 1, Client-id: 1, VRF: default
      Route Redistribution registrations:
          proto: connected
```

This example shows output of **show service-layer rib notifications registrations** command filtered by session ID for nexthop only registrations.

```
Router#show service-layer rib notifications registrations session 1 next-hops

IPv4 registrations:
  Session: 1, Client-id: 1, VRF: default
      NextHop Tracking registrations:
          192.0.2.0/32
IPv6 registrations:
  Session: 1, Client-id: 1, VRF: default
      NextHop Tracking registrations:
          2001:0DB8:1:10B::/120, Allow-Default
```

show service-layer route

To display service-layer routing information, use the **show service-layer route** command in EXEC mode.

show service-layer route { *ip-address* | *ip-address/mask* | **brief** | **client {** *client-id* | **all }** | **detail** | **ipv4** | **ipv6** { **ip-address** | **ip-address/length** | **client {** *client-id* | **all }** | **detail** } | **vrf {** *vrf-name* | **all } }**

Syntax Description

ip-address	Specify the IP address of a network to display information about.
ip-address/mask	Specifies a network with a specific prefix length.
afi-all	Displays information for both IPv4 and IPv6 addresses.
multicast	Displays multicast routes.
safi-all	Displays both unicast and multicast routes
brief	Displays the brief information about service-layer routes.
ipv4	Displays route information for IPv4 addresses.
ipv6	Displays route information for IPv6 addresses.
standby	Displays standby route information.
summary	Displays a summary of all routes.
vrf *vrf-name*	Displays vrf table information for a specific vrf-name.
vrf all	Displays all the information for all vrfs.
client *client-id*	Displays service-layer client information for a specific client ID. Client ID can be in the range of 0 to 65535.
client all	Displays service-layer client information for all clients.
detail	Displays more detailed information.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.3.1	This command was introduced.
Release 24.4.1	The command was modified to include **client** and **brief** keywords. Also, **afi-all, standby** and **summary** keywords were removed.
Release 26.1.1	The command output was modified to display the route-level metric and per-path load metric.

Usage Guidelines

The Command Syntax prior to Cisco IOS XR Software Release 24.4.1 was as follows:

show service-layer route { *ip-address* | *ip-address/mask* | **afi-all { multicast** | **safi-all** | **standby** | **summary** | **unicast }** | **ipv4** | **ipv6** { **ip-address** | **ip-address/length** | **client {***client-id* | **all }** | **detail** } | **detail** | **vrf {** *vrf-name* | **all } }**

Task ID

Task ID	Operation
config-services	read

Example

This example shows output of **show service-layer route ipv4 client all detail** command.

```
Router#show service-layer route ipv4 client all detail


VRF: default, Client 20


100.2.1.1/32, tag: 0, distance: 0, metric: 1
    path: 1, 102.12.1.2 (nexthop in vrf default) via Bundle-Ether1201
        load metric: 1, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 52
    path: 2, 102.12.2.2 (nexthop in vrf default) via Bundle-Ether1202
        load metric: 1, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 52
    path: 3, 102.12.3.2 (nexthop in vrf default) via Bundle-Ether1203
        load metric: 2, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 52
    path: 4, 102.12.4.2 (nexthop in vrf default) via Bundle-Ether1204
        load metric: 3, path flags: 0
        id: 0, protected bitmap: 0x0
        ref count: 52
  Client:20, Session:0, Operation ID:8985821339679570675
  RIB:Programmed, FIB:Unavailable, Ack Type:RIB
  Object Version:4

M=Multicast,Q=Qualified,GM=Grandmaster

Interface         Transport Address                   Priority1 State
Gi0/2/0/0         IPv4      192.168.172.122           13        M,Q
                  IPv4      192.168.172.123           17        M
Gi0/2/0/1         IPv6      fe80::2b0:4aff:fe6b:f4fc  1         Q,GM
                  IPv6      fe80::2b0:4aff:fe6b:1234  18        Q
Gi0/3/0/0         Ethernet  00b0.4a6b.f4fc
```