

SRv6 Path MTU Discovery on IOS XE devices

This document describes SRv6 Path MTU Discovery on IOS XE devices.

- Feature Information SRv6 Path MTU Discovery , on page 1
- SRv6 Path MTU Discovery, on page 2
- Restrictions for SRv6 path MTU discovery, on page 4
- Configure SRv6 encapsulation MTU, on page 4
- Enable Transit-Node PTB relay, on page 4
- Verify SRv6 path MTU discovery, on page 5
- Supported SRv6 PMTU Commands, on page 6

Feature Information SRv6 Path MTU Discovery

The table below provides the feature history for SRv6 Path MTU Discovery.

Table 1:

Feature	Release	Description
SRv6 Path MTU Discovery	Cisco IOS XE 17.18.1a	This feature introduces a mechanism to determine the maximum transmission unit (MTU) for packets traversing an SRv6 underlay network. It ensures efficient packet forwarding by preventing fragmentation and packet drops, thereby allowing network devices to dynamically adjust packet sizes to avoid exceeding link MTU limits. The system relays ICMP Packet Too Big (PTB) messages from the SRv6 underlay to the IPv6/IPv4 overlay network, supporting both Transit-node and Headend-node PTB relay methods.

SRv6 Path MTU Discovery

SRv6 path MTU discovery is a mechanism that:

- addresses challenges in determining the maximum transmission unit (MTU) for packets traversing an SRv6 underlay network,
- ensures efficient packet forwarding by preventing fragmentation and packet drops, and
- helps network devices dynamically adjust packet sizes to avoid exceeding link MTU limits.

IPv6 fragmentation happens only at the source node, and intermediate nodes do not fragment packets. When an SRv6 ingress node encapsulates a packet that is too big, it drops the packet and sends an ICMPv6 Packet Too Big (PTB) message to the source. In the SRv6 L3VPN case, when the ICMPv6 PTB message reaches the head-end ingress PE, the PE node cannot forward the message to the actual source. This is because the PE cannot determine the correct VRF context from the ICMPv6 packet.

SRv6 PMTU discovery architecture

SRv6 PMTU discovery architecture is a design that:

- ensures compatibility with both SRv6 and non-SRv6 devices within the SRv6 core.
- supports interoperability across multiple vendors.
- enables relays ICMPv6 PTB messages through the network.

Relaying ICMP PTB messages

Relaying ICMP PTB messages is a process that:

- forwards ICMPv6 PTB error messages from the SRv6 underlay network to the original packet source,
- allows the source device to adjust its packet size to prevent further fragmentation or drops, and
- ensures proper communication of MTU limitations across different network segments.

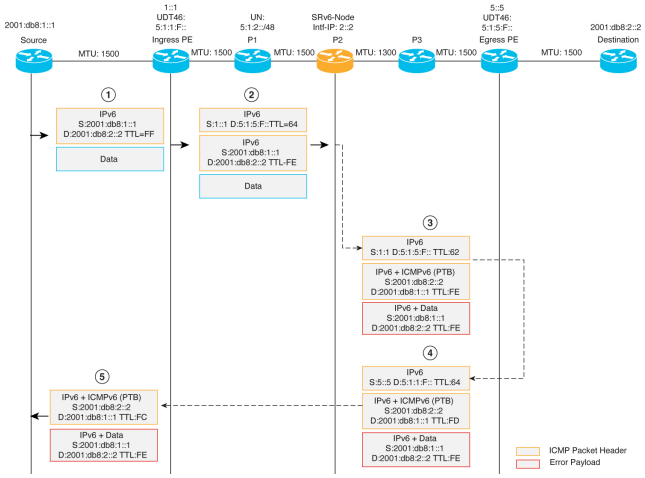
This process is essential because SRv6 does not assign a specific source address per VRF or per SRv6 policy, making it difficult for the ingress SRv6 device to identify the correct VRF to relay the ICMPv6 PTB message directly back to the source.

Transit-node PTB relay

Transit-node PTB relay is an implementation that:

- supports interoperability with IOS-XR devices,
- sends an ICMPv6 PTB message to the source device on the overlay network, and
- forwards the ICMP packet towards the egress PE using the received SRv6 outer header.

When a packet's MTU exceeds the link's MTU on an intermediate node, the intermediate node detects SRv6 encapsulation, calculates the PMTU value, generates the appropriate ICMP error message and then swaps the inner source and destination IP addresses before forwarding the packet towards the egress PE.

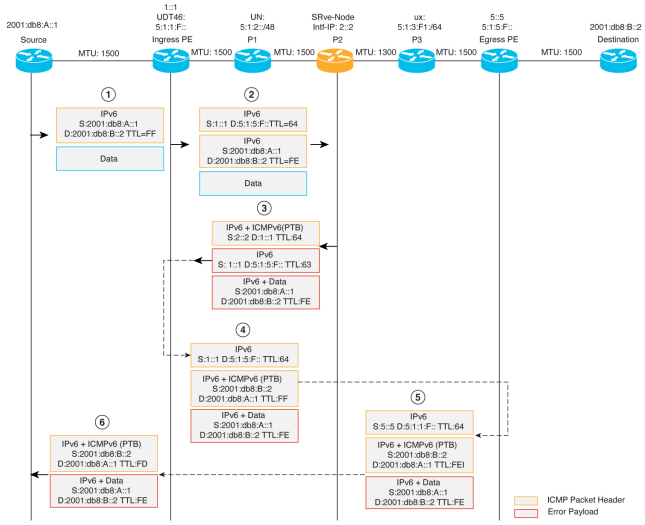


Headend-node PTB relay

Headend-node PTB relay is an implementation that

- supports intermediate devices that are non-SRv6 or SRv6 devices from other vendors,
- generates a new ICMPv6 PTB packet based on the inner header of the received ICMP PTB error message, and
- sends this new ICMP PTB message to the source device on the overlay network.

When a packet's MTU exceeds the link's MTU on an intermediate node, non-SRv6 nodes may send an ICMP PTB message towards the SRv6 source address (the SRv6 ingress PE). The SRv6 ingress PE then processes this message, generates a new ICMP PTB packet, and forwards it towards the egress PE using the SRv6 outer header from the received ICMP PTB message.



Restrictions for SRv6 path MTU discovery

The Transit-node PTB relay feature is supported only in a pure SRv6 deployment. It is not supported when the network includes a mix of IPv6-only nodes.

Configure SRv6 encapsulation MTU

Configuring SRv6 encapsulation MTU is a process that:

- involves setting the MTU value to include the SRv6 overhead,
- requires the MTU value to be based on your network requirements, and
- aims to prevent fragmentation and packet drops.

SRv6 encapsulation MTU command

Use the following command to configure the MTU:

```
segment-routing srv6
                   encapsulation
                   mtu <1280-9960>
```



Note If you prefer not to relay ICMP messages dynamically back to the source, use this command to send the ICMP message directly from the head-end. For IPv6 traffic and IPv4 traffic with the DF flag set, the head-end sends the ICMP message. For IPv4 traffic without the DF flag set, the head-end fragments the IPv4 packet before adding the SRv6 encapsulation.

Enable Transit-Node PTB relay

Enabling Transit-Node-PTB-Relay is a configuration process that:

- overrides the default Headend-Node-PTB-Relay behavior,
- activates the Transit-Node-PTB-Relay functionality, and
- serves interoperability purposes.

Headend-Node-PTB-Relay is the default behavior implemented.

Configuration command

Use the following configuration to enable Transit-Node-PTB-Relay:

```
segment-routing srv6
                   transit-node-ptb-relay
```

Verify SRv6 path MTU discovery

To provide commands and example outputs for verifying the platform's support, operational status, and counters of SRv6 Path MTU Discovery modes.

Verify platform support for SRv6 Path MTU Discovery modes

You can verify the platform's support for SRv6 Path MTU Discovery modes using the **show srv6 capabilities-parameters** command.

```
PE1#show srv6 capabilities-parameters
Platform Capabilities:
 SRv6:Yes
 PFP:Yes
 TILFA:Yes
 Transit-Node-PTB-Relay, Headend-Node-PTB-Relay
 Endpoint behaviors:
  uN (PSP/USD)
  uA (PSP/USD)
  uDT6
  uDT4
  uDT46
 Headend behaviors:
  H.ENCAP.RED
 Encap Parameters:
  Max-SL :16 (H-ENCAP-MSD:16 SL-MSD:16, END-POP-MSD:16 END-ENCAP-MSD:16)
  Encap :Collapsed
  Hop-limit propagate :Yes
  Traffic-class propagate :Yes
  Path-MTU supported :static
  SRH mode :
   combine :Yes
   stack :No
   insert :Yes

Parameters in-use:
 Encap Parameters:
  Source Address: C2:1::1, Loopback0 (Default)
  Hop-Limit: 64 (Default)
  Traffic-class: 0 (Default)
  MTU:  0 (Default IPv4 MTU)
```

Verify SRv6 Path MTU counters for Transit-Node PTB Relay

The following output shows SRv6 Path MTU counters for Transit-Node PTB Relay.

```
P4#show srv6 pmtu counters
 PMTU Transit-Relay is                                Enabled
 PMTU Headend-Remote-Relay is              Enabled
 PMTU Headend-Local-Relay is                  Enabled
 PMTU SRv6 not enabled:                          16
 PMTU Handle ICMP6 PTB Calls:              11
 Matched Block for SRv6 Encap:                11
 ICMP6 Transit PTB Gen:                           1
 ICMP4 Transit PTB Gen:                           3
 IPv6 MTU change needed:                        1
 IPv4 MTU change needed:                       10
 ICMP4 Internal ICMP Build Error:               2
 IPv4 DF bit is not set:                                5
 PMTU Transit-Relay disabled hit:             15
```

```
 IPv6 Last MTU: 1360 Src: 1700::7 Dst: 6800::8 Originator 1411::4 Transit [06-16 00:52:01.236]

 IPv6 Min  MTU: 1360 Src: 1700::7 Dst: 6800::8 Originator 1411::4 Transit [06-16 00:52:01.236]

 IPv4 Last MTU: 1360 Src: 17.0.0.7 Dst: 68.0.0.8 Originator 1411::4 Transit DF  [06-16
00:51:45.966]
 IPv4 Min  MTU: 1360 Src: 17.0.0.7 Dst: 68.0.0.8 Originator 1411::4 Transit [06-16
00:51:09.587]
 SRv6 FIB Queue Size:             100
 SRv6 FIB Queue Peak:             1
 SRv6 FIB Queue Drops:            0
 SRv6 FIB Queue Enqueues:         11
```

Verify SRv6 Path MTU counters for Headend-Node PTB Relay

The following output shows SRv6 Path MTU counters for Headend-Node PTB Relay.

```
PE1#show srv6 pmtu counters
 PMTU Transit-Relay is                                Disabled
 PMTU Headend-Remote-Relay is               Enabled
 PMTU Headend-Local-Relay is                   Enabled
 PMTU Handle ICMP6 PTB Calls:               11
 Found SRH for SRv6 Encap:                      11
 ICMP6 Headend-Local PTB Gen:               1
 ICMP4 Headend-Local PTB Gen:               3
 IPv6 MTU change needed:                         1
 IPv4 MTU change needed:                         10
 ICMP4 Internal ICMP Build Error:                2
 IPv4 DF bit is not set:                                  5
 PMTU counters cleared:                                4
 IPv6 Last MTU: 1336 Src: 1700::7 Dst: 6800::8 Originator A001::1 Headend-Local VRF: vpn-101
 [06-15 14:26:31.229]
 IPv6 Min  MTU: 1336 Src: 1700::7 Dst: 6800::8 Originator A001::1 Headend-Local VRF: vpn-101
 [06-15 14:26:31.229]
 IPv4 Last MTU: 1336 Src: 17.0.0.7 Dst: 68.0.0.8 Originator A001::1 Headend-Local VRF:
vpn-101 DF  [06-15 14:26:21.635]
 IPv4 Min  MTU: 1336 Src: 17.0.0.7 Dst: 68.0.0.8 Originator A001::1 Headend-Local VRF:
vpn-101 [06-15 14:26:03.760]
 SRv6 FIB Queue Size:             100
 SRv6 FIB Queue Peak:             1
 SRv6 FIB Queue Drops:            0
 SRv6 FIB Queue Enqueues:         11
```

Supported SRv6 PMTU Commands

You can manage and troubleshoot SRv6 PMTU features using the following commands:

Table 2: SRv6 PMTU Commands

Command	Description
clear segment-routing counters pmtu	Clears SRv6 PMTU statistics.
debug srv6 pmtu	Enables debugging for SRv6 PMTU.
show srv6 capabilities-parameters	Verifies a platform's support for SRv6 Path MTU Discovery.

Command	Description
show srv6 pmtu counters	Displays SRv6 Path MTU counters.