

MPLS LDP and SR MPLS to provide VPN using Inter AS option C (IOS-XR)

Contents

[Introduction](#)

[Prerequisite](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Problem](#)

[Solution](#)

[SR Global Block](#)

[SR Mapping Server](#)

[Mapping Client](#)

[Topology – Inter-AS Option C \(with Route-Reflectors\)](#)

[Brief Overview of the Topology Diagram](#)

[IP Addressing Schema](#)

[Configurations](#)

[Related Information](#)

Introduction

This document describes how a partial SR deployment can be used to provide SR benefits to LDP-based traffic including a possible application of SR in the context of inter-domain MPLS use-cases.

Prerequisite

Requirements

Cisco recommends that you have knowledge of these topics:

- Segment Routing
- MPLS
- Inter-AS
- LDP

Components Used

The information in this document is based on these software and hardware versions:

- Supporting Segment Routing
- NCS5500, ASR9000

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

This document outlines the mechanisms through which SR interworks with LDP in cases where a mix of SR-capable and non-SR-capable routers co-exist within the same network and more precisely in the same routing domain.

A Multicast Control Plane Client (MCC), operating at a node, must ensure that the incoming label it installs in the MPLS data plane of Node has been uniquely allocated and Segment Routing makes use of the Segment Routing Global Block (SRGB) for the label allocation. The use of the SRGB allows SR to co-exist with any other MCC.

Tip: This information helps you with the Inter-AS solution for MPLS SR and MPLS LDP using Inter-AS Option C.

Problem

SR deployment in a heterogeneous environment with SR MPLS Control plane interoperate with MPLS LDP Control Plane using Inter-AS option C defined in RFC 4364.

This document describes a method by which Service provider Network using MPLS LDP and SR MPLS to provide Virtual Private Network using Inter-AS option C.

Solution

A brief overview of Inter-AS Option C:

Inter-AS Option C is the third option for interconnecting multi-AS backbones covered in RFC 4364. It's the most scalable option of the three so far and it has its own applicability scenarios that we must be aware of to apply this design properly.

Option C is a good candidate because it is scalable. ASBR's don't carry any VPN routes and they just take care of the distribution of labeled IPv4 routes of the PE's within their own AS.

To improve scalability, one MP-EBGP VPNv4 session transports all VPN routes (external routes) between PEs or RR. In the case of using RR to exchange the external routes, the next-hop of the VPNv4 routes must be preserved.

The ASBR uses EBGP to exchange the internal PE routing information between AS (internal routes). These internal routes correspond to the BGP next-hops of the external routes advertised through the multi-hop MP-EBGP session between PEs or RRs. The internal routes advertised by the ASBRs can be used to establish the MP-EBGP sessions between PEs and allows for LSP setup from the ingress to the egress PE.

Option C is a very good solution from the scalability point of view and is the way to go for the same SP multi-AS networks.

A brief overview of Segment Routing:

Segment Routing (SR) leverages the source routing and tunneling paradigms. A node steers a packet through a controlled set of instructions, called segments, by prepending the packet with an SR header. A segment can represent any instruction, topological or service-based. SR allows enforcing a flow through any topological path and service chain while maintaining a per-flow state only at the ingress node of the SR domain. The Segment Routing architecture can be directly applied to the MPLS data plane with a little change on the forwarding plane. It requires minor extensions to the existing link-state routing protocols. Segment Routing can also be applied to IPv6 with a new type of routing extension header.

A segment is encoded as an MPLS label. An ordered list of segments is encoded as a stack of labels. The segment to process is on the top of the stack. Upon completion of a segment, the related label is popped from the stack. Segment-routing MPLS data-plane operations are push, swap, and pop as per the traditional MPLS forwarding. Following segment types are defined in Segment Routing:

- Prefix Segment
- Adjacency Segment
- BGP Peering Segment
- BGP Prefix Segment

SR Global Block

The Segment Routing Global Block (SRGB) is the range of label values preserved for segment routing in the LSD. The SRGB label values are assigned as prefix segment identifiers (SIDs) to SR-enabled nodes and have global significance throughout the domain.

- SR label range can NOT start below 16,000. The default SR global block is : 16,000 - 24,000.
- SRGB configuration is NOT address-family specific because the “SR-Capabilities Sub-TLV” of router capability TLV defined in is not address-family specific.
- If CLI results in enlarging or moving the default SRGB, then it is OK to require a reload but only if there are clients who have labels in the new range.
- Before configuring SRGB, the administrator needs to make sure that a portion of the label base that is being configured for Segment-Routing is free and is not being used by any other MPLS LSD clients.

SR Mapping Server

Mapping Server centrally assigns prefix-SIDs for some or all of the known prefixes. A router must be able to act as a mapping server, mapping client, or both.

- A mapping server is a control plane mechanism and its position is comparable to a BGP Route-reflector
- Allows the user to configure non-overlapping SID mapping entries to specify the prefix-SIDs for some or all prefixes.
- ISIS advertises the local SID-mapping policy in 'SID/Label Binding TLV'
- The mapping server must be resilient, redundancy should be provided

Main functions of Mapping Server include:

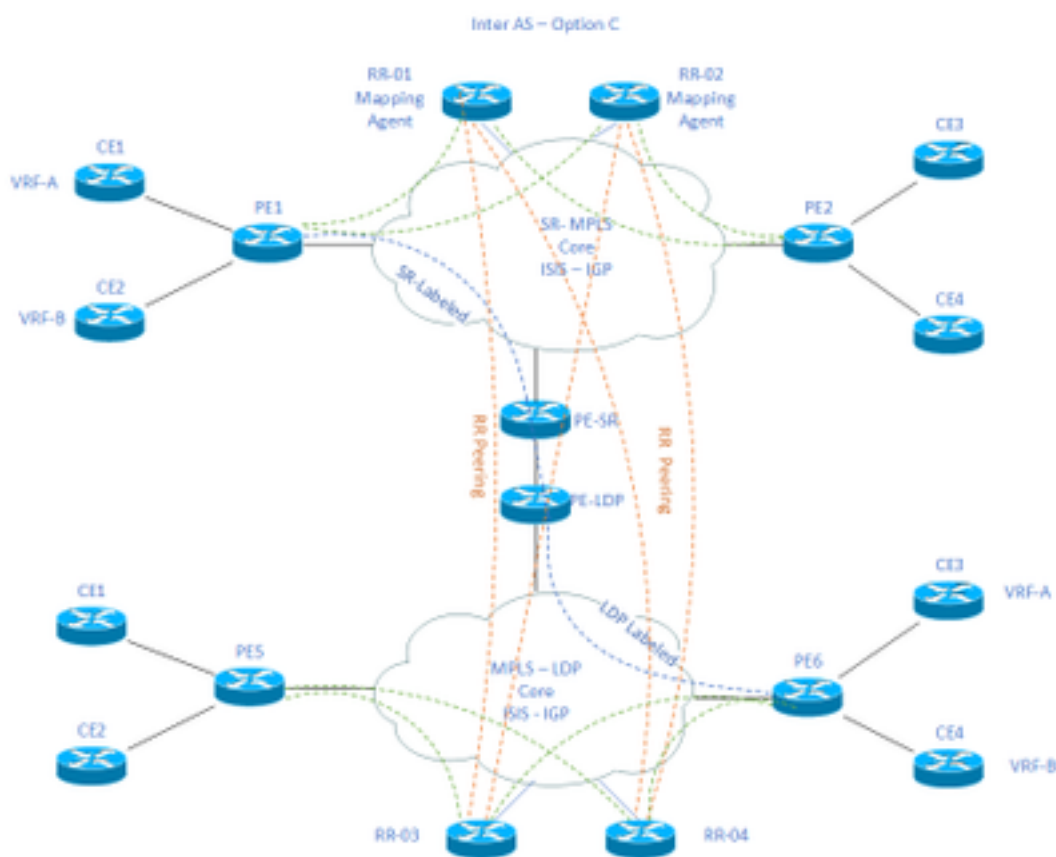
- Advertise Prefix-to-SID mappings in IGP on behalf of other non-SR-capable nodes prefix-to-sid mappings are configured on the Mapping Server
- Enable SR-capable nodes to interwork with (non-SR-capable) LDP nodes, a Mapping Server is required for SR/LDP interworking

Mapping Client

- Receives and parses remotely received SID/Label Binding TLV to create remote SID-mapping entries. Using the remotely learned and locally configured mapping entries, construct the non-overlapping consistent active mapping policy.
- IGP instance uses the active mapping policy to (re)calculate the prefix-SIDs of some or all prefixes.

This section helps you to understand & configuring L3 Virtual Private Network (VPN) service between Provider network with SR Capable Network peering/Connecting with Non-SR capable Network. In this section, you learn how to Configure Option C Defined in RFC “4364” and Use Cases.

Topology – Inter-AS Option C (with Route-Reflectors)



Brief Overview of the Topology Diagram

Towards the top of the topology diagram, we have an SR-enabled network comprising of Route-reflectors, Provider Edge routers, and Customer Edge routers.

The customer edge routers CE1 and CE2 have VRF A and VRF B respectively. This belongs to AS 65002.

Towards the bottom of the topology diagram, we have an LDP-enabled network comprising of Route-reflectors, Provider Edge routers, and customer edge routers.

The customer edge routers CE3 and CE4 have VRF A and VRF B respectively. This belongs to AS65001.

VRF A and VRF B on either end of the SR and LDP enabled networks to need to communicate with each other.

IP Addressing Schema

Hostname	IP Address
RR-01	10.0.0.1
RR-02	10.0.0.2
PE1	10.0.0.3
PE2	10.0.0.4
CE1	10.0.0.5
CE2	10.0.0.6
RR-03	10.0.1.1
RR-04	10.0.1.2
PE5	10.0.1.3
PE6	10.0.1.4
CE3	10.0.1.5
CE4	10.0.1.6

Configurations

The configurations of the devices are described.

RR-1

```
segment-routing mpls
!
mapping-server
!
prefix-sid-map
  address-family ipv4
    10.0.0.1/32 index 200 range 10
```

```
    exit-address-family
!
!

interface Loopback0

    description Loopback0

    ip address 10.0.0.1 255.255.255.255

    ip router isis 65002
!

!

router isis 65002

    net xx.xxxx.xxxx.xxxx.xx

    metric-style wide

    segment-routing mpls

    segment-routing prefix-sid-map advertise-local
!

router bgp 65002

    bgp router-id 10.0.0.1

    neighbor 10.0.0.3 remote-as 65002

    neighbor 10.0.0.3 description rr client

    neighbor 10.0.0.3 update-source Loopback0

    neighbor 10.0.0.4 remote-as 65002

    neighbor 10.0.0.4 description rr client

    neighbor 10.0.0.4 update-source Loopback0

    neighbor 10.0.0.2 remote-as 65002

    neighbor 10.0.0.2 description iBGP peer

    neighbor 10.0.0.2 update-source Loopback0

    neighbor 10.0.1.1 remote-as 65001

    neighbor 10.0.1.1 ebgp-multihop 255

    neighbor 10.0.1.1 update-source Loopback0
```

```
neighbor 10.0.1.2 remote-as 65001
neighbor 10.0.1.2 ebgp-multihop 255
neighbor 10.0.1.2 update-source Loopback0
!
address-family ipv4
neighbor 10.0.0.3 activate
neighbor 10.0.0.3 route-reflector-client
neighbor 10.0.0.4 activate
neighbor 10.0.0.4 route-reflector-client
neighbor 10.0.0.2 activate
neighbor 10.0.1.1 activate
neighbor 10.0.1.2 activate
exit-address-family
!
address-family vpnv4
neighbor 10.0.0.3 activate
neighbor 10.0.0.3 send-community extended
neighbor 10.0.0.3 route-reflector-client
neighbor 10.0.0.4 activate
neighbor 10.0.0.4 send-community extended
neighbor 10.0.0.4 route-reflector-client
neighbor 10.0.0.2 activate
neighbor 10.0.0.2 send-community extended
neighbor 10.0.1.1 activate
neighbor 10.0.1.1 send-community both
neighbor 10.0.1.1 next-hop-unchanged
neighbor 10.0.1.2 activate
neighbor 10.0.1.2 send-community both
neighbor 10.0.1.2 next-hop-unchanged
exit-address-family
!
```

RR-2

```
segment-routing mpls
!
mapping-server
!
prefix-sid-map
  address-family ipv4
    10.0.0.2/32 index 200 range 10
  exit-address-family
!
!

interface Loopback0
  description Loopback0
  ip address 10.0.0.2 255.255.255.255
  ip router isis 65002
!

!

router isis 65002
  net xx.xxxx.xxxx.xxxx.xx
  metric-style wide
  segment-routing mpls
  segment-routing prefix-sid-map advertise-local
!

router bgp 65002
  bgp router-id 10.0.0.2
  neighbor 10.0.0.3 remote-as 65002
  neighbor 10.0.0.3 description rr client
  neighbor 10.0.0.3 update-source Loopback0
  neighbor 10.0.0.4 remote-as 65002
```



```
neighbor 10.0.0.4 description rr client
neighbor 10.0.0.4 update-source Loopback0
neighbor 10.0.0.1 remote-as 65002
neighbor 10.0.0.1 description iBGP peer
neighbor 10.0.0.1 update-source Loopback0
neighbor 10.0.1.1 remote-as 65001
neighbor 10.0.1.1 ebgp-multihop 255
neighbor 10.0.1.1 update-source Loopback0
neighbor 10.0.1.2 remote-as 65001
neighbor 10.0.1.2 ebgp-multihop 255
neighbor 10.0.1.2 update-source Loopback0
!
address-family ipv4
neighbor 10.0.0.3 activate
neighbor 10.0.0.3 route-reflector-client
neighbor 10.0.0.4 activate
neighbor 10.0.0.4 route-reflector-client
neighbor 10.0.0.1 activate
neighbor 10.0.1.1 activate
neighbor 10.0.1.2 activate
exit-address-family
!
address-family vpnv4
neighbor 10.0.0.3 activate
neighbor 10.0.0.3 send-community extended
neighbor 10.0.0.3 route-reflector-client
neighbor 10.0.0.4 activate
neighbor 10.0.0.4 send-community extended
neighbor 10.0.0.4 route-reflector-client
neighbor 10.0.0.1 activate
neighbor 10.0.0.1 send-community extended
neighbor 10.0.1.1 activate
```

```
neighbor 10.0.1.1 send-community both
neighbor 10.0.1.1 next-hop-unchanged
neighbor 10.0.1.2 activate
neighbor 10.0.1.2 send-community both
neighbor 10.0.1.2 next-hop-unchanged
exit-address-family
```

```
!
PE-1
```

```
interface Loopback0
description Loopback0
ip address 10.0.0.3 255.255.255.255
ip router isis 65002
```

```
!
```

```
vrf A # Define VRF A
```

```
address-family ipv4 unicast
```

```
import route-target
```

```
65000:1
```

```
!
```

```
export route-target
```

```
65000:1
```

```
!
```

```
!
```

```
vrf B # Define VRF B
```

```
address-family ipv4 unicast
```

```
import route-target
```

```
65000:2
```

```
!
```

```
export route-target
```

```
65000:2
```

```
!
```

```
!
```

```
router isis 65002 # ISIS Level 2
```

```
is-type level-2-only
```

```
net xx.xxxx.xxxx.xxxx.xx
```

```
address-family ipv4 unicast
```

```
metric-style wide
```

```
advertise link attributes
```

```
mpls traffic-eng level-2-only
```

```
mpls traffic-eng router-id Loopback0
```

```
router-id Loopback0
```

```
segment-routing mpls sr-prefer
```

```
!
```

```
router bgp 65002 # BGP
```

```
bgp router-id 10.0.0.3
```

```
neighbor-group RR
```

```
remote-as 65002
```

```
update-source Loopback0
```

```
address-family vpnv4 unicast
```

```
!
```

```
!
```

```
neighbor 10.0.0.1
```

```
use neighbor-group RR
```

```
!
```

```
neighbor 10.0.0.2
```

```
use neighbor-group RR
```

```
vrf A

rd 65000:1

address-family ipv4 unicast

  redistribute connected

  allocate-label all

!

neighbor 10.0.0.5          # IP address of CE1

  remote-as 61001

  ebgp-multihop 255

  update-source Loopback100

  address-family ipv4 unicast

!

vrf B

rd 65000:2

address-family ipv4 unicast

  redistribute connected

  allocate-label all

!

neighbor 10.0.0.6          # IP address of CE2

  remote-as 61001

  ebgp-multihop 255

  update-source Loopback101

  address-family ipv4 unicast

!

interface GigabitEthernet1      # Link to CE-01

vrf A

  ipv4 address x.x.x.x 255.255.255.0

!

interface GigabitEthernet2      # Link to CE-02

vrf B

  ipv4 address x.x.x.x 255.255.255.0

!
```

```
segment-routing
```

```
global-block 16000 23999
```

```
!
```

RR-3

```
interface Loopback0
```

```
description Loopback0
```

```
ip address 10.0.1.1 255.255.255.255
```

```
ip router isis 65001
```

```
!
```

```
!
```

```
router isis 65001
```

```
net xx.xxxx.xxxx.xxxx.xx
```

```
metric-style wide
```

```
segment-routing mpls
```

```
segment-routing prefix-sid-map advertise-local
```

```
!
```

```
router bgp 65001
```

```
bgp router-id 10.0.1.1
```

```
neighbor 10.0.1.3 remote-as 65001
```

```
neighbor 10.0.1.3 description rr client
```

```
neighbor 10.0.1.3 update-source Loopback0
```

```
neighbor 10.0.1.4 remote-as 65001
```

```
neighbor 10.0.1.4 description rr client
```

```
neighbor 10.0.1.4 update-source Loopback0
```

```
neighbor 10.0.1.2 remote-as 65001
```

```
neighbor 10.0.1.2 description iBGP peer
```

```
neighbor 10.0.1.2 update-source Loopback0
```

```
neighbor 10.0.0.1 remote-as 65002
```

```
neighbor 10.0.0.1 ebgp-multihop 255
neighbor 10.0.0.1 update-source Loopback0
neighbor 10.0.0.2 remote-as 65002
neighbor 10.0.0.2 ebgp-multihop 255
neighbor 10.0.0.2 update-source Loopback0
!
address-family ipv4
neighbor 10.0.1.3 activate
neighbor 10.0.1.3 route-reflector-client
neighbor 10.0.1.4 activate
neighbor 10.0.1.4 route-reflector-client
neighbor 10.0.1.2 activate
neighbor 10.0.0.1 activate
neighbor 10.0.0.2 activate
exit-address-family
!
address-family vpnv4
neighbor 10.0.1.3 activate
neighbor 10.0.1.3 send-community extended
neighbor 10.0.1.3 route-reflector-client
neighbor 10.0.1.4 activate
neighbor 10.0.1.4 send-community extended
neighbor 10.0.1.4 route-reflector-client
neighbor 10.0.1.2 activate
neighbor 10.0.1.2 send-community extended
neighbor 10.0.0.1 activate
neighbor 10.0.0.1 send-community both
neighbor 10.0.0.1 next-hop-unchanged
neighbor 10.0.0.2 activate
neighbor 10.0.0.2 send-community both
neighbor 10.0.0.2 next-hop-unchanged
exit-address-family
```

!

RR-4

```
interface Loopback0
  description Loopback0
  ip address 10.0.1.2 255.255.255.255
  ip router isis 65001
!

!

router isis 65001
  net xx.xxxx.xxxx.xxxx.xx
  metric-style wide
  segment-routing mpls
  segment-routing prefix-sid-map advertise-local
!

router bgp 65001
  bgp router-id 10.0.1.2
  neighbor 10.0.1.3 remote-as 65001
  neighbor 10.0.1.3 description rr client
  neighbor 10.0.1.3 update-source Loopback0
  neighbor 10.0.1.4 remote-as 65001
  neighbor 10.0.1.4 description rr client
  neighbor 10.0.1.4 update-source Loopback0
  neighbor 10.0.1.1 remote-as 65001
  neighbor 10.0.1.1 description iBGP peer
  neighbor 10.0.1.1 update-source Loopback0
  neighbor 10.0.0.1 remote-as 65002
```

```
neighbor 10.0.0.1 ebgp-multihop 255
neighbor 10.0.0.1 update-source Loopback0
neighbor 10.0.0.2 remote-as 65002
neighbor 10.0.0.2 ebgp-multihop 255
neighbor 10.0.0.2 update-source Loopback0
!
address-family ipv4
neighbor 10.0.1.3 activate
neighbor 10.0.1.3 route-reflector-client
neighbor 10.0.1.4 activate
neighbor 10.0.1.4 route-reflector-client
neighbor 10.0.1.1 activate
neighbor 10.0.0.1 activate
neighbor 10.0.0.2 activate
exit-address-family
!
address-family vpnv4
neighbor 10.0.1.3 activate
neighbor 10.0.1.3 send-community extended
neighbor 10.0.1.3 route-reflector-client
neighbor 10.0.1.4 activate
neighbor 10.0.1.4 send-community extended
neighbor 10.0.1.4 route-reflector-client
neighbor 10.0.1.1 activate
neighbor 10.0.1.1 send-community extended
neighbor 10.0.0.1 activate
neighbor 10.0.0.1 send-community both
neighbor 10.0.0.1 next-hop-unchanged
neighbor 10.0.0.2 activate
neighbor 10.0.0.2 send-community both
neighbor 10.0.0.2 next-hop-unchanged
exit-address-family
```


!

PE-6

```
interface Loopback0
  description Loopback0
  ip address 10.0.1.3 255.255.255.255
  ip router isis 65001
```

!

```
vrf A # Define VRF A
  address-family ipv4 unicast
```

```
  import route-target
```

```
    65000:1
```

!

```
  export route-target
```

```
    65000:1
```

!

```
vrf B # Define VRF B
  address-family ipv4 unicast
```

```
  import route-target
```

```
    65000:2
```

!

```
  export route-target
```

```
    65000:2
```

!

```
router isis 65001
  is-type level-2-only
```

```
net xx.xxxx.xxxx.xxxx.xx
address-family ipv4 unicast
metric-style wide
advertise link attributes
mpls traffic-eng level-2-only
mpls traffic-eng router-id Loopback0
router-id Loopback0
segment-routing mpls sr-prefer
!
```

```
router bgp 65001
  bgp router-id 10.0.1.3
  neighbor-group RR
  remote-as 65002
  update-source Loopback0
  address-family vpnv4 unicast
```

```
!
```

```
!
```

```
neighbor 10.0.1.1
  use neighbor-group RR
!
neighbor 10.0.1.2
  use neighbor-group RR
```

```
vrf A
  rd 65000:1
  address-family ipv4 unicast
    redistribute connected
    allocate-label all
  !
```

```
neighbor 10.0.1.5          # IP address of CE3
```

```
remote-as 61001

ebgp-multihop 255

update-source Loopback100

address-family ipv4 unicast

!

vrf B

rd 65000:2

address-family ipv4 unicast

redistribute connected

allocate-label all

!

neighbor 10.0.1.6          # IP address of CE4

remote-as 61001

ebgp-multihop 255

update-source Loopback101

address-family ipv4 unicast

!

interface GigabitEthernet1      # Link to CE3

vrf A

ipv4 address x.x.x.x 255.255.255.0

!

interface GigabitEthernet2      # Link to CE4

vrf B

ipv4 address x.x.x.x 255.255.255.0

!
```

Related Information

- [Technical Support & Documentation - Cisco Systems](#)
- <https://tools.ietf.org/html/rfc8661>
- <https://tools.ietf.org/html/rfc4659>
- <https://tools.ietf.org/html/rfc4364>
- <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-ldp-interop-08>
- <https://tools.ietf.org/html/draft-ietf-isis-segment-routing-extensions-19>
- <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-ldp-interop-15#ref-I-D.ietf-isis-segment-routing-extensions>

- <https://learningnetwork.cisco.com/s/question/0D53i00000Ksqy9CAB/interas-option-c>