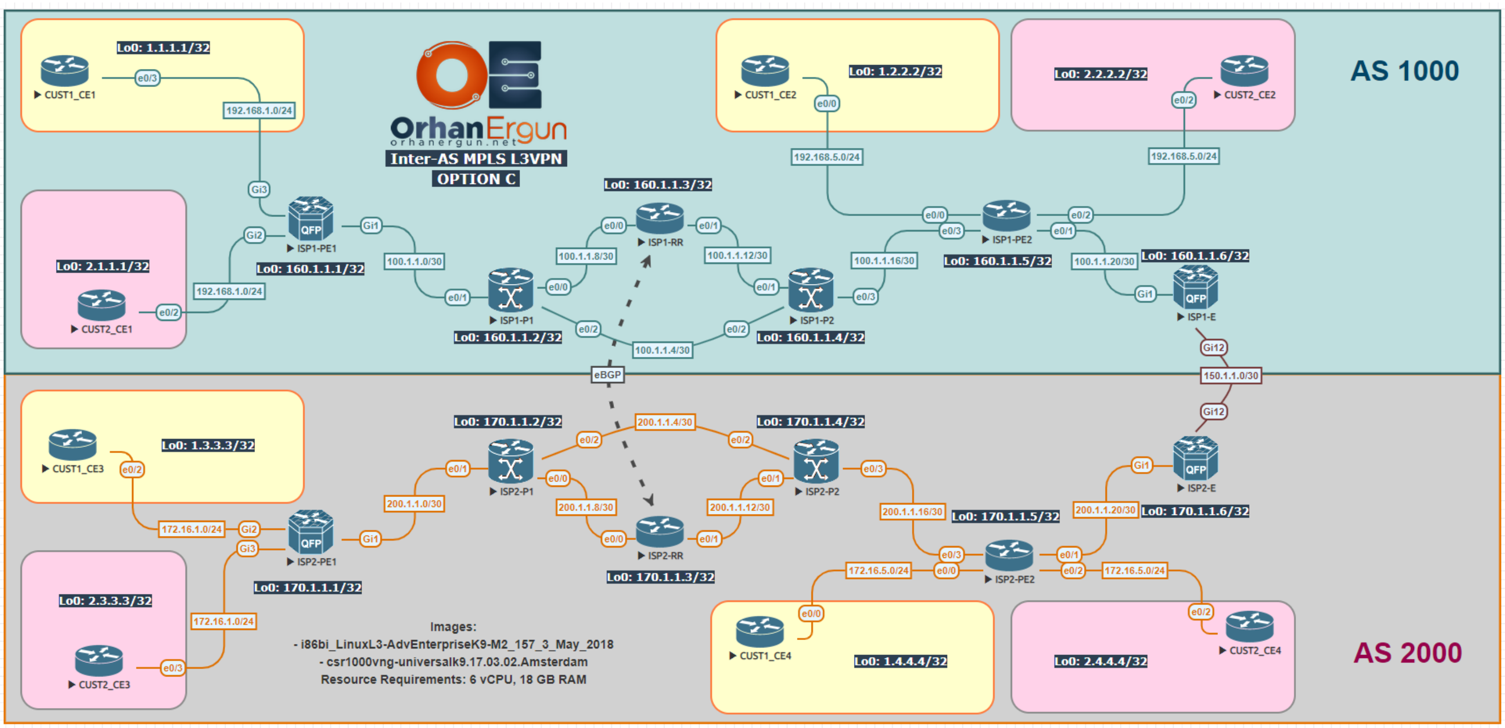
Inter-AS MPLS L3VPN Option C

Topology:



BGP AF/SAF: VPNv4 Unicast

PE-CE Routing Protocols:

- OSPF (Cust1)
- EIGRP Named-Mode (Cust2)

Task 01:

- Provide MPLS L3VPN service to the customers (Cust1 PE-CE: OSPF, Cust2 PE-CE: EIGRP)
- ISP1-RR and ISP2-RR should be configured as the Route-Reflectors

Solution:

You are already familiar with the L3VPN configuration, so we are not going to go through detailed explanation of the configuration steps:

ISP1-PE1:

```
vrf definition CUST1
 rd 1000:1
 !
 address-family ipv4
  route-target export 160.1.1.1:1
  route-target import 160.1.1.5:1
  route-target import 160.1.1.6:1
 exit-address-family
!
!
interface GigabitEthernet3
 vrf forwarding CUST1
 ip address 192.168.1.1 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1 vrf CUST1
 redistribute bgp 1000
 network 0.0.0.0 255.255.255.255 area 0
!
!
router bgp 1000
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
```

```
 exit-address-family
!
vrf definition CUST2
 rd 1000:2
 !
 address-family ipv4
  route-target export 160.1.1.1:2
  route-target import 160.1.1.5:2
  route-target import 160.1.1.6:2
 exit-address-family
!
!
interface GigabitEthernet2
 vrf forwarding CUST2
 ip address 192.168.1.1 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid
!
router eigrp CUST2
!
address-family ipv4 unicast vrf CUST2 autonomous-system 2
  !
  topology base
   redistribute bgp 1000
  exit-af-topology
  network 0.0.0.0
 exit-address-family
!
router bgp 1000
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
```

```
 exit-address-family
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 160.1.1.3 remote-as 1000
 neighbor 160.1.1.3 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 160.1.1.3 activate
  neighbor 160.1.1.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
```

ISP1-RR:

```
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 160.1.1.1 remote-as 1000
 neighbor 160.1.1.1 update-source Loopback0
 neighbor 160.1.1.5 remote-as 1000
 neighbor 160.1.1.5 update-source Loopback0
 no neighbor 160.1.1.6 remote-as 1000
```

```
 no neighbor 160.1.1.6 update-source Loopback0
 !
 address-family vpnv4
  neighbor 160.1.1.1 activate
  neighbor 160.1.1.1 send-community extended
  neighbor 160.1.1.1 route-reflector-client
  neighbor 160.1.1.5 activate
  neighbor 160.1.1.5 send-community extended
  neighbor 160.1.1.5 route-reflector-client
  neighbor 160.1.1.6 activate
  neighbor 160.1.1.6 send-community extended
  neighbor 160.1.1.6 route-reflector-client
 exit-address-family
```

ISP1-PE2:

```
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 160.1.1.3 remote-as 1000
 neighbor 160.1.1.3 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 160.1.1.3 activate
  neighbor 160.1.1.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
 !
```

```
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
!
vrf definition CUST1
 rd 1000:1
 !
 address-family ipv4
  route-target export 160.1.1.5:1
  route-target import 160.1.1.1:1
  route-target import 160.1.1.6:1
 exit-address-family
!
!
interface Ethernet0/0
 vrf forwarding CUST1
 ip address 192.168.5.1 255.255.255.0
 duplex auto
!
router ospf 1 vrf CUST1
 redistribute bgp 1000 subnets
 network 0.0.0.0 255.255.255.255 area 0
!
!
router bgp 1000
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
!
vrf definition CUST2
 rd 1000:2
 !
```

```
 address-family ipv4
  route-target export 160.1.1.5:2
  route-target import 160.1.1.1:2
  route-target import 160.1.1.6:2
 exit-address-family
!
!
interface Ethernet0/2
 vrf forwarding CUST2
 ip address 192.168.5.1 255.255.255.0
 duplex auto
!
router eigrp CUST2
!
address-family ipv4 unicast vrf CUST2 autonomous-system 2
  !
  topology base
   redistribute bgp 1000
  exit-af-topology
  network 0.0.0.0
 exit-address-family
!
router bgp 1000
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
!
```

Service Provider 1 basic configuration is done.

We will configure the ISP2 the same way:

ISP2-PE1:

```
router bgp 2000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 170.1.1.3 remote-as 2000
 neighbor 170.1.1.3 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 170.1.1.3 activate
  neighbor 170.1.1.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
vrf definition CUST1
 rd 2000:1
 !
 address-family ipv4
  route-target export 170.1.1.1:1
  route-target import 170.1.1.5:1
  route-target import 170.1.1.6:1
 exit-address-family
!
!
interface GigabitEthernet2
 vrf forwarding CUST1
 ip address 172.16.1.1 255.255.255.0
```

```
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1 vrf CUST1
 redistribute bgp 2000
 network 0.0.0.0 255.255.255.255 area 0
!
!
router bgp 2000
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
!
vrf definition CUST2
 rd 2000:2
 !
 address-family ipv4
  route-target export 170.1.1.1:2
  route-target import 170.1.1.5:2
  route-target import 170.1.1.6:2
 exit-address-family
!
!
interface GigabitEthernet3
 vrf forwarding CUST2
 ip address 172.16.1.1 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid
!
router eigrp CUST2
!
```

```
address-family ipv4 unicast vrf CUST2 autonomous-system 2
  !
  topology base
   redistribute bgp 2000
  exit-af-topology
  network 0.0.0.0
 exit-address-family
!
router bgp 2000
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
!
```

ISP2-RR:

```
router bgp 2000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 170.1.1.1 remote-as 2000
 neighbor 170.1.1.1 update-source Loopback0
 neighbor 170.1.1.5 remote-as 2000
 neighbor 170.1.1.5 update-source Loopback0
 no neighbor 170.1.1.6 remote-as 2000
 no neighbor 170.1.1.6 update-source Loopback0
 !
 address-family vpnv4
  neighbor 170.1.1.1 activate
  neighbor 170.1.1.1 send-community extended
  neighbor 170.1.1.1 route-reflector-client
  neighbor 170.1.1.5 activate
  neighbor 170.1.1.5 send-community extended
  neighbor 170.1.1.5 route-reflector-client
  neighbor 170.1.1.6 activate
```

```
  neighbor 170.1.1.6 send-community extended
  neighbor 170.1.1.6 route-reflector-client
 exit-address-family
```

ISP2-PE2:

```
router bgp 2000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 170.1.1.3 remote-as 2000
 neighbor 170.1.1.3 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 170.1.1.3 activate
  neighbor 170.1.1.3 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
vrf definition CUST1
 rd 2000:1
 !
 address-family ipv4
  route-target export 170.1.1.5:1
  route-target import 170.1.1.1:1
  route-target import 170.1.1.6:1
 exit-address-family
```

```
!
!
interface Ethernet0/0
 vrf forwarding CUST1
 ip address 172.16.5.1 255.255.255.0
 duplex auto
!
router ospf 1 vrf CUST1
 redistribute bgp 2000 subnets
 network 0.0.0.0 255.255.255.255 area 0
!
!
router bgp 2000
 !
 address-family ipv4 vrf CUST1
  redistribute ospf 1 match internal external 1 external 2
 exit-address-family
!
vrf definition CUST2
 rd 2000:2
 !
 address-family ipv4
  route-target export 170.1.1.5:2
  route-target import 170.1.1.1:2
  route-target import 170.1.1.6:2
 exit-address-family
!
!
interface Ethernet0/2
 vrf forwarding CUST2
 ip address 172.16.5.1 255.255.255.0
 duplex auto
!
router eigrp CUST2
```

```
!
address-family ipv4 unicast vrf CUST2 autonomous-system 2
  !
  topology base
   redistribute bgp 2000
  exit-af-topology
  network 0.0.0.0
 exit-address-family
!
router bgp 2000
 !
 address-family ipv4 vrf CUST2
  redistribute eigrp 2
 exit-address-family
!
```

ISP2 basic configuration is also done.

So the customer sites inside each ISP have end to end reachability.

But what about customer sites reachability between two different sites which are connected to the different ISPs?

They don't have any connectivity so far (for example CUST1_CE1 cannot ping CUST1_CE3).

The solution is using Inter-AS MPLS VPN different options.

There are 4 option: A, B, C and D.

The difference between those options are mostly how Edge routers interact with each other at the edge of the network (also in option C we will discuss about RR interaction on different Service Providers).

Task 02:

- Make sure that all customer sites have IP reachability (Use Option C)

Solution:

In order to understand how Option C works, please complete the Option A and B labs first.

In terms of basic configuration the only difference between Option A, B with C is: RRs do need need to form VPNv4 unicast neighborship with ASBRs (ISP1-E and ISP2-E). we highlighted that part in the above basic configurations.

- In option A: ASBRs was forming eBGP IPv4 neighborship for each customer under a separate VRF.
- In Option B: ASBRs was forming MP-BGP VPNv4 unicast neighborship with each other and there was no VRF definition for each customer on ASBRs. In addition to that, ASBRs was forming IPv4 Labeled-Unicast neighborship in order to label-switch the packet between each other.
- In Option C: ASBRs are going to only form IPv4 Labeled-Unicast with each other to label-switch the packet and also advertise PE devices and RRs IP addresses to each other.

It is an interesting solution (Option C), in this option, Route-Reflectors on each Service Provider are forming VPNv4 unicast neighborship with each other, in fact the session running between them is an eBGP session. It comes to your mind that we need to also do next-hop-self between them otherwise there is a chance of sub-optimal routing.

Let's do the configuration and I will explane how it works and what cavieats you need to take into the consideration:

```
ISP1-E:
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 150.1.1.2 remote-as 2000
 !
 address-family ipv4
  network 160.1.1.1 mask 255.255.255.255
  network 160.1.1.3 mask 255.255.255.255
  network 160.1.1.5 mask 255.255.255.255
  neighbor 150.1.1.2 activate
```

```
  neighbor 150.1.1.2 send-label
 exit-address-family
!
interface GigabitEthernet12
 ip address 150.1.1.1 255.255.255.252
 mpls bgp forwarding
!

ISP2-E:
router bgp 2000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 150.1.1.1 remote-as 1000
 !
 address-family ipv4
  network 170.1.1.1 mask 255.255.255.255
  network 170.1.1.3 mask 255.255.255.255
  network 170.1.1.5 mask 255.255.255.255
  neighbor 150.1.1.1 activate
  neighbor 150.1.1.1 send-label
 exit-address-family
!
interface GigabitEthernet12
 ip address 150.1.1.2 255.255.255.252
 mpls bgp forwarding
!
```

What an easy configuration on ASBRs! The only thing that needs to be done is to form IPv4 Labeled-Unicast neighborship between them and leak PE and RR IP addresses.

Those BGP learned IP addresses needs to be also advertised into IGP (If there is a Unified MPLS, there is no need to do redistribution into IGP and vise versa. But in this case we are not using seamless/Unified MPLS. So We will do redistribution from BGP into IGP. This step is necessary, because the internal routers need to know how to reach to those prefixes. For example ISP1-RR needs to know about the LSP (Label switched path) towards ISP2-RR.

Let's do the IGP redistribution on ASBRs:

```
ISP1-E:
router ospf 1000
 redistribute bgp 1000 metric 2500 tag 150111
!

ISP2-E:
router ospf 1000
 redistribute bgp 2000 metric 2500 tag 150112
!
```

Verification:

```
ISP1-RR#show mpls forwarding-table 170.1.1.3
Local      Outgoing    Prefix           Bytes Label   Outgoing    Next Hop
Label      Label       or Tunnel Id     Switched      interface
3010       4009        170.1.1.3/32     0             Et0/1       100.1.1.14

ISP2-RR#show mpls forwarding-table 160.1.1.3
Local      Outgoing    Prefix           Bytes Label   Outgoing    Next Hop
Label      Label       or Tunnel Id     Switched      interface
3001       4009        160.1.1.3/32     0             Et0/1       200.1.1.14

ISP2-RR#trace 160.1.1.3 probe 1 source lo 0
Type escape sequence to abort.
Tracing the route to 160.1.1.3
VRF info: (vrf in name/id, vrf out name/id)
  1 200.1.1.14 [MPLS: Label 4009 Exp 0] 3 msec
  2 200.1.1.18 [MPLS: Label 5014 Exp 0] 1 msec
  3 200.1.1.22 [MPLS: Label 6012 Exp 0] 4 msec
  4 150.1.1.1 [MPLS: Label 6002 Exp 0] 2 msec
  5 100.1.1.21 [MPLS: Label 5001 Exp 0] 2 msec
  6 100.1.1.17 [MPLS: Label 4005 Exp 0] 1 msec
  7 100.1.1.13 1 msec
```

Everything works fine, there is an LSP between two RRs Loopback addresses, so we are ready to configure VPNv4 Unicast neighborship between Route-Reflectors:

```
ISP1-RR:
router bgp 1000
 neighbor 170.1.1.3 remote-as 2000
 neighbor 170.1.1.3 ebgp-multihop 255
 neighbor 170.1.1.3 update-source Loopback0
 !
 address-family vpnv4
  neighbor 170.1.1.3 activate
  neighbor 170.1.1.3 send-community extended
  neighbor 170.1.1.3 next-hop-unchanged
 exit-address-family

ISP2-RR:
router bgp 2000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 160.1.1.3 remote-as 1000
 neighbor 160.1.1.3 ebgp-multihop 255
 neighbor 160.1.1.3 update-source Loopback0
 !
 address-family vpnv4
  neighbor 160.1.1.3 activate
  neighbor 160.1.1.3 send-community extended
  neighbor 160.1.1.3 next-hop-unchanged
 exit-address-family
```

They should form VPNv4 unicast neighborship and advertise the local routes to each other:

```
ISP1-RR#show bgp vpnv4 uni all summary | begin Neigh
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
160.1.1.1       4         1000      38      49       17    0    0 00:27:58        4
160.1.1.5       4         1000      39      44       17    0    0 00:30:13        4
170.1.1.3       4         2000      42      41       17    0    0 00:27:28        8
```

ISP1-RR has formed the neighborship with ISP2-RR and it received 8 prefixes (VPNv4 prefixes) from the neighboring RR:

```
ISP1-RR#show bgp vpnv4 uni all | begin Net
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 10002000:1
 *>i  1.1.1.1/32       160.1.1.1                2    100      0 ?
 *>i  1.2.2.2/32       160.1.1.5               11    100      0 ?
 *>   1.3.3.3/32       170.1.1.1                              0 2000 ?
 *>   1.4.4.4/32       170.1.1.5                              0 2000 ?
 *>   172.16.1.0/24    170.1.1.1                              0 2000 ?
 *>   172.16.5.0/24    170.1.1.5                              0 2000 ?
 *>i  192.168.1.0      160.1.1.1                0    100      0 ?
 *>i  192.168.5.0      160.1.1.5                0    100      0 ?
Route Distinguisher: 10002000:2
 *>i  2.1.1.1/32       160.1.1.1            10880    100      0 ?
 *>i  2.2.2.2/32       160.1.1.5          1024640    100      0 ?
 *>   2.3.3.3/32       170.1.1.1                              0 2000 ?
 *>   2.4.4.4/32       170.1.1.5                              0 2000 ?
 *>   172.16.1.0/24    170.1.1.1                              0 2000 ?
 *>   172.16.5.0/24    170.1.1.5                              0 2000 ?
 *>i  192.168.1.0      160.1.1.1                0    100      0 ?
 *>i  192.168.5.0      160.1.1.5                0    100      0 ?
```

Take a look at 1.3.3.3/32 route, the next-hop is not changed, thanks to the next-hop-unchanged command on the Route-Reflector:

```
ISP1-PE1#show bgp vpnv4 uni rd 10002000:1 1.3.3.3/32
BGP routing table entry for 10002000:1:1.3.3.3/32, version 19
Paths: (1 available, best #1, table CUST1)
  Not advertised to any peer
  Refresh Epoch 2
  2000
    170.1.1.1 (metric 2500) (via default) from 160.1.1.3 (160.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: OSPF DOMAIN ID:0x0005:0x000000010200 RT:170.1.1.1:1
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:172.16.1.1:0
      mpls labels in/out nolabel/1010
      rx pathid: 0, tx pathid: 0x0
      Updated on Apr 11 2021 12:08:59 UTC
```

```
ISP1-PE1#show ip cef vrf CUST1 1.3.3.3/32
1.3.3.3/32
  nexthop 100.1.1.2 GigabitEthernet1 label 2007-(local:1010) 1010


CUST1_CE1#traceroute 1.3.3.3 source lo 0
Type escape sequence to abort.
Tracing the route to 1.3.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.1.1 4 msec 1 msec 0 msec
  2 100.1.1.2 [MPLS: Labels 2007/1010 Exp 0] 3 msec 3 msec 2 msec
  3 100.1.1.6 [MPLS: Labels 4008/1010 Exp 0] 3 msec 6 msec 2 msec
  4 100.1.1.18 [MPLS: Labels 5013/1010 Exp 0] 3 msec 3 msec 2 msec
  5 100.1.1.22 [MPLS: Labels 6011/1010 Exp 0] 4 msec 3 msec 3 msec
  6 150.1.1.2 [MPLS: Labels 6004/1010 Exp 0] 3 msec 4 msec 3 msec
  7 200.1.1.21 [MPLS: Labels 5011/1010 Exp 0] 2 msec 3 msec 3 msec
  8 200.1.1.17 [MPLS: Labels 4006/1010 Exp 0] 2 msec 3 msec 2 msec
  9 200.1.1.5 [MPLS: Labels 2006/1010 Exp 0] 3 msec 3 msec 2 msec
 10 172.16.1.1 [MPLS: Label 1010 Exp 0] 4 msec 2 msec 3 msec
 11 172.16.1.100 3 msec 2 msec 3 msec
```

Interesting! VPN label stayed intact end-to-end between two PE devices in different Autonomous Systems. The VPN label is 1010 and it is not being changed until it reaches to the destination.

Let’s try removing Next-Hop-Unchanged command on RRs and see the result:

```
ISP1-RR:
router bgp 1000
 address-family vpnv4
  no neighbor 170.1.1.3 next-hop-unchanged
!
ISP2-RR:
router bgp 2000
 address-family vpnv4
  no neighbor 160.1.1.3 next-hop-unchanged
!
```

Verification:

```
ISP1-RR(config-router-af)#do clear bgp vpnv4 uni * so


CUST1_CE1#traceroute 1.3.3.3 source lo 0
Type escape sequence to abort.
Tracing the route to 1.3.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.1.1 4 msec 1 msec 0 msec
  2 100.1.1.2 [MPLS: Labels 2008/3016 Exp 0] 7 msec 3 msec 2 msec
  3 100.1.1.6 [MPLS: Labels 4009/3016 Exp 0] 3 msec 2 msec 3 msec
  4 100.1.1.18 [MPLS: Labels 5014/3016 Exp 0] 3 msec 3 msec 3 msec
  5 100.1.1.22 [MPLS: Labels 6012/3016 Exp 0] 4 msec 3 msec 3 msec
  6 150.1.1.2 [MPLS: Labels 6002/3016 Exp 0] 4 msec 3 msec 3 msec
  7 200.1.1.21 [MPLS: Labels 5001/3016 Exp 0] 3 msec 3 msec 3 msec
  8 200.1.1.17 [MPLS: Labels 4001/3016 Exp 0] 3 msec 3 msec 3 msec
  9 200.1.1.13 [MPLS: Label 3016 Exp 0] 2 msec 3 msec 3 msec
 10 200.1.1.9 [MPLS: Labels 2006/1010 Exp 0] 3 msec 3 msec 2 msec
 11 172.16.1.1 [MPLS: Label 1010 Exp 0] 4 msec 2 msec 3 msec
 12 172.16.1.100 3 msec 3 msec 3 msec
```

1 more hop this time! Packet is taking the sub-optimal path, it goes to the RR (which is not needed or necessary, this is only an offline RR, ISP1-P2 and ISP1-P1 have a direct connectivity with each other.

Let's make it even worth!

```
ISP2-RR(config-if)#int e0/0
ISP2-RR(config-if)#bandwidth 100


ISP2-P1(config-if)#int e0/0
ISP2-P1(config-if)#band 100
```

We have decreased the bandwidth of the link between ISP2-RR and ISP2-P1, the packet will be sent to the ISP2-RR and comes back to the ISP2-P2 again! One more step will be added to the packet path! You need to be aware of this when you are configuring Inter-AS MPLS VPNs using Option C, otherwise you can cause sub-optimal pat easily if you don't know how this option works:

```
CUST1_CE1#traceroute 1.3.3.3 source lo 0

Type escape sequence to abort.

Tracing the route to 1.3.3.3

VRF info: (vrf in name/id, vrf out name/id)

  1 192.168.1.1 4 msec 0 msec 1 msec

  2 100.1.1.2 [MPLS: Labels 2008/3016 Exp 0] 3 msec 3 msec 3 msec

  3 100.1.1.6 [MPLS: Labels 4009/3016 Exp 0] 3 msec 5 msec 4 msec

  4 100.1.1.18 [MPLS: Labels 5014/3016 Exp 0] 3 msec 3 msec 2 msec

  5 100.1.1.22 [MPLS: Labels 6012/3016 Exp 0] 3 msec 3 msec 3 msec

  6 150.1.1.2 [MPLS: Labels 6002/3016 Exp 0] 4 msec 3 msec 4 msec

  7 200.1.1.21 [MPLS: Labels 5001/3016 Exp 0] 2 msec 3 msec 3 msec

  8 200.1.1.17 [MPLS: Labels 4001/3016 Exp 0] 2 msec 3 msec 3 msec

  9 200.1.1.13 [MPLS: Label 3016 Exp 0] 3 msec 3 msec 3 msec

 10 200.1.1.14 [MPLS: Labels 4006/1010 Exp 0] 3 msec 3 msec 3 msec

 11 200.1.1.5 [MPLS: Labels 2006/1010 Exp 0] 3 msec 3 msec 3 msec

 12 172.16.1.1 [MPLS: Label 1010 Exp 0] 3 msec 2 msec 3 msec

 13 172.16.1.100 3 msec 3 msec 3 msec
```