Contents

Introduction
The Problem
Assert Mechanism on the Default MDT
Conclusion
Assert Mechanism with Data MDTs
Conclusion

Introduction

This document describes mVPN (Multicast Virtual Provider Network) with dual-homed Source and Data MDT (Multicast Distribution Tree). An example in Cisco IOS® is used in order to illustrate the behavior.

The Problem

If a source in the mVPN world is dual-homed to two Ingress Provider Edge (PE) routers, it could be possible for the two Ingress PE routers to both forward traffic for one (S,G) into the Multiprotocol Label Switching (MPLS) cloud. This is possible if, for example, there are two Egress PE routers and each Reverse Path Forwarding (RPF) to a different Ingress PE router. If both Ingress PE routers forward onto the Default MDT, then the assert mechanism will kick in and one Ingress PE wins the assert mechanism and the other loses so that one and only one Ingress PE continues to forward the Customer (C-) (S,G) onto the MDT. However, if for any reason the assert mechanism did not start on the Default MDT, then it is possible for both Ingress PE routers to begin to transmit the C-(S,G) multicast traffic onto one Data-MDT they initiate. Because the traffic is not on the Default MDT anymore, but on Data MDTs, both Ingress PE routers do not receive the C-(S,G) traffic from each other on the MDT/Tunnel interface. This can cause persistent duplicate traffic downstream. This document explains the solution to this problem.

Assert Mechanism on the Default MDT

The information in this section holds true for the Default MDT, regardless of the core tree protocol. The chosen core tree protocol is Protocol Independent Multicast (PIM).

Cisco IOS is used for the examples, but everything that is mentioned applies equally for Cisco IOS-XR. All multicast Groups used are Source Specific Multicast (SSM) groups.

Look at Figure 1. Dual-Homed-Source-1. There are two Ingress PE routers (PE1 and PE2) and two Egress PE routers (PE3 and PE4). The Source is at CE1 with IP address 10.100.1.6. CE1 is dual-homed to PE1 and PE2.

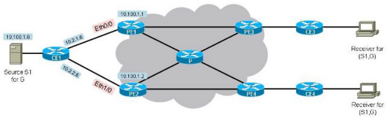

Figure 1. Dual-Homed-Source-1

The configuration on all PE routers (the Route Distinguisher (RD) can be different on the PE routers) is:

```
vrf definition one
 rd 1:1
 !
 address-family ipv4
 mdt default 232.10.10.10
 route-target export 1:1
 route-target import 1:1
 exit-address-family
!
```

In order to get both Ingress PE routers to start to forward the multicast stream (10.100.1.6,232.1.1.1) out onto the Default MDT, they must both receive a Join from an Egress PE. Look at the topology in Figure1. Dual-Homed-Source-1. You can see that by default, if all the costs of the edge links are the same and all of the costs of the core links are the same, then PE3 will RPF towards PE1 and PE4 will RPF towards PE2 for (10.100.1.6,232.1.1.1). They both RPF to their closest Ingress PE. This output confirms this:

```
PE3#show ip rpf vrf one 10.100.1.6
RPF information for ? (10.100.1.6)
 RPF interface: Tunnel0
 RPF neighbor: ? (10.100.1.1)
 RPF route/mask: 10.100.1.6/32
 RPF type: unicast (bgp 1)
 Doing distance-preferred lookups across tables
 BGP originator: 10.100.1.1
 RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE3 has RPF to PE1.

```
PE4#show ip rpf vrf one 10.100.1.6
RPF information for ? (10.100.1.6)
 RPF interface: Tunnel0
 RPF neighbor: ? (10.100.1.2)
 RPF route/mask: 10.100.1.6/32
 RPF type: unicast (bgp 1)
 Doing distance-preferred lookups across tables
 BGP originator: 10.100.1.2
 RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE4 has RPF to PE2. The reason that PE3 picks PE1 as the RPF neighbor is that the unicast route towards 10.100.1.6/32 in Virtual Routing/Forwarding (VRF) one is the best via PE1. PE3 actually receives the route 10.100.1.6/32 from both PE1 and PE2. All the criteria in the Border Gateway Protocol (BGP) Best Path Calculation Algorithm are the same, except for the cost towards the BGP next-hop address.

```
PE3#show bgp vpnv4 unicast vrf one  10.100.1.6/32
BGP routing table entry for 1:3:10.100.1.6/32, version 333
Paths: (2 available, best #1, table one)
 Advertised to update-groups:
    21
 Refresh Epoch 1
 Local, imported path from 1:1:10.100.1.6/32 (global)
   10.100.1.1 (metric 11) (via default) from 10.100.1.5 (10.100.1.5)
     Origin incomplete, metric 11, localpref 100, valid, internal,best
     Extended Community: RT:1:1 OSPF DOMAIN ID:0x0005:0x000000640200
       OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:10.2.4.1:0
     Originator: 10.100.1.1, Cluster list: 10.100.1.5
     Connector Attribute: count=1
      type 1 len 12 value 1:1:10.100.1.1
     mpls labels in/out nolabel/32
     rx pathid: 0, tx pathid: 0x0
 Refresh Epoch 1
 Local, imported path from 1:2:10.100.1.6/32 (global)
   10.100.1.2 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
     Origin incomplete, metric 11, localpref 100, valid, internal
     Extended Community: RT:1:1 OSPF DOMAIN ID:0x0005:0x000000640200
       OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:10.2.2.2:0
     Originator: 10.100.1.2, Cluster list: 10.100.1.5
     Connector Attribute: count=1
      type 1 len 12 value 1:2:10.100.1.2
     mpls labels in/out nolabel/29
     rx pathid: 0, tx pathid: 0 PE4#show bgp vpnv4 unicast vrf one  10.100.1.6/32
BGP routing table entry for 1:4:10.100.1.6/32, version 1050
Paths: (2 available, best #2, table one)
 Advertised to update-groups:
    2
 Refresh Epoch 1
 Local, imported path from 1:1:10.100.1.6/32 (global)
   10.100.1.1 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
     Origin incomplete, metric 11, localpref 100, valid, internal
     Extended Community: RT:1:1 OSPF DOMAIN ID:0x0005:0x000000640200
       OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:10.2.4.1:0
     Originator: 10.100.1.1, Cluster list: 10.100.1.5
     Connector Attribute: count=1
      type 1 len 12 value 1:1:10.100.1.1
     mpls labels in/out nolabel/32
     rx pathid: 0, tx pathid: 0
 Refresh Epoch 1
 Local, imported path from 1:2:10.100.1.6/32 (global)
   10.100.1.2 (metric 11) (via default) from 10.100.1.5 (10.100.1.5)
     Origin incomplete, metric 11, localpref 100, valid, internal, best
     Extended Community: RT:1:1 OSPF DOMAIN ID:0x0005:0x000000640200
       OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:10.2.2.2:0
     Originator: 10.100.1.2, Cluster list: 10.100.1.5
     Connector Attribute: count=1
      type 1 len 12 value 1:2:10.100.1.2
     mpls labels in/out nolabel/29
     rx pathid: 0, tx pathid: 0x0
```

The best path chosen by PE3 is the path advertised by PE1 because that has the lowest Interior Gateway Protocol (IGP) cost (11), versus the IGP cost (21) towards PE2. For PE4 it is the reverse. The topology reveals that from PE3 to PE1 there is only one hop, while from PE3 to PE2 there are two hops. Since all links have the same IGP cost, PE3 picks the path from PE1 as the best.

The Multicast Routing Information Base (MRIB) for (10.100.1.6,232.1.1.1) looks like this on PE1 and PE2 when there is no multicast traffic yet:

```
PE1#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:00:12/00:03:17, flags: sT
 Incoming interface: Ethernet0/0, RPF nbr 10.2.1.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:00:12/00:03:17 PE2#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:00:47/00:02:55, flags: sT
 Incoming interface: Ethernet1/0, RPF nbr 10.2.2.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:00:47/00:02:55
```

PE1 and PE2 both received a PIM Join for (10.100.1.6,232.1.1.1). The Tunnel0 interface is in the Outgoing Interface List (OIL) for the multicast entry on both routers.

The multicast traffic starts to flow for (10.100.1.6,232.1.1.1). "Debug ip pim vrf one 232.1.1.1" and "debug ip mrouting vrf one 232.1.1.1" show us that the arrival of multicast traffic onto Tunnel0 (in the OIL) of both Ingress PE routers, causes the assert mechanism to run.

PE1

```
PIM(1): Send v2 Assert on Tunnel0 for 232.1.1.1, source 10.100.1.6, metric [110/11]
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
MRT(1): not RPF interface, source address 10.100.1.6, group address 232.1.1.1
PIM(1): Received v2 Assert on Tunnel0 from 10.100.1.2
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PIM(1): We lose, our metric [110/11]
PIM(1): Prune Tunnel0/232.10.10.10 from (10.100.1.6/32, 232.1.1.1)
MRT(1): Delete Tunnel0/232.10.10.10 from the olist of (10.100.1.6, 232.1.1.1)
MRT(1): Reset the PIM interest flag for (10.100.1.6, 232.1.1.1)
MRT(1): set min mtu for (10.100.1.6, 232.1.1.1) 1500->18010 - deleted
PIM(1): Received v2 Join/Prune on Tunnel0 from 10.100.1.3, not to us
PIM(1): Join-list: (10.100.1.6/32, 232.1.1.1), S-bit set
```

PE2

```
PIM(1): Received v2 Assert on Tunnel0 from 10.100.1.1
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PIM(1): We win, our metric [110/11]
PIM(1): (10.100.1.6/32, 232.1.1.1) oif Tunnel0 in Forward state
PIM(1): Send v2 Assert on Tunnel0 for 232.1.1.1, source 10.100.1.6, metric [110/11]
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PIM(1): Received v2 Join/Prune on Tunnel0 from 10.100.1.3, to us
PIM(1): Join-list: (10.100.1.6/32, 232.1.1.1), S-bit set
PIM(1): Update Tunnel0/10.100.1.3 to (10.100.1.6, 232.1.1.1), Forward state, by PIM SG Join
```

If the metric and distance is the same of both routers towards the Source 10.100.1.6, then there is a tie-breaker in order to determine the assert winner. The tie-breaker is the highest IP address of the PIM neighbor on the Tunnel0 (Default MDT). In this case, this is PE2:

```
PE1#show ip pim vrf one neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
     P - Proxy Capable, S - State Refresh Capable, G - GenID Capable,
     L - DR Load-balancing Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
10.100.1.4        Tunnel0                  06:27:57/00:01:29 v2    1 / DR S P G
10.100.1.3        Tunnel0                  06:28:56/00:01:24 v2    1 / S P G
10.100.1.2        Tunnel0                  06:29:00/00:01:41 v2    1 / S P G PE1#show ip pim vrf
one interface

Address          Interface                Ver/   Nbr    Query  DR         DR
                                         Mode   Count  Intvl  Prior
10.2.1.1         Ethernet0/0              v2/S   0      30     1          10.2.1.1
10.2.4.1         Ethernet1/0              v2/S   0      30     1          10.2.4.1
10.100.1.1       Lspvif1                  v2/S   0      30     1          10.100.1.1
10.100.1.1       Tunnel0                  v2/S   3      30     1          10.100.1.4
```

PE1 removed Tunnel0 from the OIL of the multicast entry because of the asserts. Since the OIL became empty, the multicast entry is pruned.

```
PE1#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:17:24/00:00:01, flags: sPT
 Incoming interface: Ethernet0/0, RPF nbr 10.2.1.6
 Outgoing interface list: Null
```

PE2 has the A-flag set on the interface Tunnel0, because it is the assert winner.

```
PE2#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:17:20/00:02:54, flags: sT
 Incoming interface: Ethernet1/0, RPF nbr 10.2.2.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:17:20/00:02:54, A
```

PE2 periodically sends an assert on Tunnel0 (Default MDT), just before the assert timer expires. As such PE2 remains the assert winner.

```
PE2#
PIM(1): Send v2 Assert on Tunnel0 for 232.1.1.1, source 10.100.1.6, metric [110/11]
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
```

Conclusion

The assert mechanism also works with a Tunnel interface in the OIL. The asserts are exchanged over the Default MDT when the Ingress PE routers receive C-(S,G) multicast traffic on the associated Tunnel interface which is in the OIL.

Assert Mechanism with Data MDTs

Most of the time when Data MDTs are configured, the assert mechanism will still run on the Default MDT as the C-(S,G) traffic is only switched over from the Default MDT to the Data MDTs after three seconds. Then the same occurs as previously described. Note that **there is only one tunnel interface per multicast-enabled VRF**: the Default MDT and all Data MDTs use one tunnel interface only. This tunnel interface is used in the OIL on the Ingress PE routers or as an RPF interface on the Egress PE routers.

In some cases it is possible that the assert mechanism is not triggered before the Data MDTs are signaled. Then it is possible that the C-(S,G) multicast traffic starts to be forwarded on a Data MDT on both Ingress PE routers PE1 and PE2. In such cases, this could lead to permanent duplicate C-(S,G) multicast traffic across the MPLS core network. In order to avoid this, this solution was implemented: when an Ingress PE router sees another Ingress PE router announce a Data MDT for which the PE router is also an Ingress PE router, it joins that Data MDT. In principle, only Egress PE routers (that have a downstream receiver) would join the Data MDT. Because Ingress PE routers join the Data MDT announced by other Ingress PE routers, it leads to the Ingress PE router receiving multicast traffic from the Tunnel interface which is present in the OIL, and hence this triggers the assert mechanism and leads to one of the Ingress PE routers to stop forwarding the C-(S,G) multicast traffic onto its Data MDT (with the Tunnel interface), while the other Ingress PE (the assert winner) can continue to forward the C-(S,G) multicast traffic onto its Data MDT.

For the next example, assume that the Ingress PE routers PE1 and PE2 never saw the C-(S,G) multicast traffic from each other on the Default MDT. The traffic is on the Default MDT for only three seconds and it is not difficult to understand that this can occur if there is, for example, temporary traffic loss on the core network.

The configuration for Data MDT is added to all PE routers. The configuration on all PE routers (the RD can be different on the PE routers) is:

```
vrf definition one
 rd 1:1
 !
 address-family ipv4
 mdt default 232.10.10.10
mdt data 232.11.11.0 0.0.0.0
 route-target export 1:1
 route-target import 1:1
 exit-address-family
!
```

As soon as PE1 and PE2 see the traffic from the Source, they create a C-(S,G) entry. Both Ingress PE routers forward the C-(S,G) multicast traffic onto the Default MDT. Egress PE routers PE3 and PE4 receive the multicast traffic and forward it. Due to a temporary issue, PE2 does not see the traffic from PE1 and vice-versa on the Default MDT. They both send a Data MDT Join Type Length Value (TLV) out on the Default MDT.

If there is no C-(S,G) traffic, you see this multicast state on the Ingress PE routers:

```
PE1#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:00:45/00:02:44, flags: sT
 Incoming interface: Ethernet0/0, RPF nbr 10.2.1.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:00:45/00:02:42 PE2#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:02:18/00:03:28, flags: sT
 Incoming interface: Ethernet1/0, RPF nbr 10.2.2.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:02:18/00:03:28
```

The y-flag is not yet set. Both Ingress PE routers have the Tunnel0 interface in the OIL. This is due to the fact that PE3 has RPF towards PE1 and PE4 has RPF towards PE2 for C-(S,G).

When multicast traffic for C-(S,G) starts to flow, both PE1 and PE2 forward the traffic. The threshold for Data MDT is crossed on both Ingress PE routers and both send out a Data MDT Join TLV and after three seconds start forwarding onto their Data MDT. Notice that PE1 joins the Data MDT sourced by PE2 and PE2 joins the Data MDT sourced by PE1.

```
PE1#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:01:26/00:03:02, flags: sTy
 Incoming interface: Ethernet0/0, RPF nbr 10.2.1.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:01:26/00:03:02 PE2#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
      L - Local, P - Pruned, R - RP-bit set, F - Register flag,
      T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report,
      Z - Multicast Tunnel, z - MDT-data group sender,
      Y - Joined MDT-data group, y - Sending to MDT-data group,
      G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
      N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
      Q - Received BGP S-A Route, q - Sent BGP S-A Route,
      V - RD & Vector, v - Vector, p - PIM Joins on route,
      x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:00:41/00:02:48, flags: sTy
 Incoming interface: Ethernet1/0, RPF nbr 10.2.2.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:00:41/00:02:48
```

Both PE1 and PE receive traffic for C-(S,G) on the Tunnel0 interface (but now from the Data MDT, not the Default MDT) and the assert mechanism kicks in. Only PE2 continues to forward the C-(S,G) traffic on its Data MDT:

```
PE1#
PIM(1): Send v2 Assert on Tunnel0 for 232.1.1.1, source 10.100.1.6, metric [110/11]
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
MRT(1): not RPF interface, source address 10.100.1.6, group address 232.1.1.1
PIM(1): Received v2 Assert on Tunnel0 from 10.100.1.2
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PIM(1): We lose, our metric [110/11]
PIM(1): Prune Tunnel0/232.11.11.0 from (10.100.1.6/32, 232.1.1.1)
MRT(1): Delete Tunnel0/232.11.11.0 from the olist of (10.100.1.6, 232.1.1.1)
MRT(1): Reset the PIM interest flag for (10.100.1.6, 232.1.1.1)
```

```
PIM(1): MDT Tunnel0 removed from (10.100.1.6,232.1.1.1)
MRT(1): Reset the y-flag for (10.100.1.6,232.1.1.1)
PIM(1): MDT next_hop change from: 232.11.11.0 to 232.10.10.10 for (10.100.1.6, 232.1.1.1)
Tunnel0
MRT(1): set min mtu for (10.100.1.6, 232.1.1.1) 1500->18010 - deleted
PIM(1): MDT threshold dropped for (10.100.1.6,232.1.1.1)
PIM(1): Receive MDT Packet (9889) from 10.100.1.2 (Tunnel0), length (ip: 44, udp: 24), ttl: 1
PIM(1): TLV type: 1 length: 16 MDT Packet length: 16 PE2#
PIM(1): Received v2 Assert on Tunnel0 from 10.100.1.1
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PIM(1): We win, our metric [110/11]
PIM(1): (10.100.1.6/32, 232.1.1.1) oif Tunnel0 in Forward state
PIM(1): Send v2 Assert on Tunnel0 for 232.1.1.1, source 10.100.1.6, metric [110/11]
PIM(1): Assert metric to source 10.100.1.6 is [110/11]
PE2#
PIM(1): Received v2 Join/Prune on Tunnel0 from 10.100.1.3, to us
PIM(1): Join-list: (10.100.1.6/32, 232.1.1.1), S-bit set
PIM(1): Update Tunnel0/10.100.1.3 to (10.100.1.6, 232.1.1.1), Forward state, by PIM SG Join
MRT(1): Update Tunnel0/232.10.10.10 in the olist of (10.100.1.6, 232.1.1.1), Forward state - MAC
built
MRT(1): Set the y-flag for (10.100.1.6,232.1.1.1)
PIM(1): MDT next_hop change from: 232.10.10.10 to 232.11.11.0 for (10.100.1.6, 232.1.1.1)
Tunnel0
```

PE1 no longer has the tunnel interface in the OIL.

```
PE1#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:10:23/00:00:04, flags: sPT
 Incoming interface: Ethernet0/0, RPF nbr 10.2.1.6
 Outgoing interface list: Null
```

PE2 has the A-flag set on the Tunnel0 interface:

```
PE2#show ip mroute vrf one 232.1.1.1 10.100.1.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
```

```
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 00:10:00/00:02:48, flags: sTy
 Incoming interface: Ethernet1/0, RPF nbr 10.2.2.6
 Outgoing interface list:
   Tunnel0, Forward/Sparse, 00:08:40/00:02:48, A
```

Conclusion

The assert mechanism also works when Data MDTs are used. The asserts are exchanged over the Default MDT when the Ingress PE routers receive C-(S,G) multicast traffic on the associated Tunnel interface which is in the OIL.