

# MDT de datos y origen de dos direcciones en mVPN

## Contenido

[Introducción](#)

[El problema](#)

[Mecanismo de afirmación en el MDT predeterminado](#)

[Conclusión](#)

[Mecanismo de afirmación con MDT de datos](#)

[Conclusión](#)

## Introducción

Este documento describe mVPN (Red de proveedor virtual de multidifusión) con origen de doble conexión y MDT de datos (Árbol de distribución de multidifusión). Se utiliza un ejemplo en Cisco IOS<sup>®</sup> para ilustrar el comportamiento.

## El problema

Si un origen en el mundo de mVPN tiene un doble enlace con dos routers periféricos de proveedor de entrada (PE), los dos routers PE de entrada podrían reenviar tráfico para uno (S,G) a la nube de switching de etiquetas multiprotocolo (MPLS). Esto es posible si, por ejemplo, hay dos routers PE de salida y cada Reenvío de ruta inversa (RPF) a un router PE de entrada diferente. Si ambos routers PE de ingreso se reenvían a la MDT predeterminada, el mecanismo de aserción se iniciará y un PE de ingreso gana el mecanismo de aserción y el otro pierde de modo que uno y sólo un PE de ingreso continúa reenviando al Cliente (C-) (S,G) a la MDT. Sin embargo, si por alguna razón el mecanismo de aserción no se inició en el MDT predeterminado, entonces es posible que ambos routers PE de ingreso empiecen a transmitir el tráfico multicast C-(S,G) en un Data-MDT que inician. Debido a que el tráfico ya no está en el MDT predeterminado, sino en los MDT de datos, ambos routers PE de entrada no reciben el tráfico C-(S,G) entre sí en la interfaz MDT/Túnel. Esto puede causar un tráfico duplicado persistente en sentido descendente. Este documento explica la solución a este problema.

## Mecanismo de afirmación en el MDT predeterminado

La información de esta sección es verdadera para el MDT predeterminado, independientemente del protocolo del árbol principal. El protocolo de árbol de núcleo elegido es multidifusión independiente del protocolo (PIM).

Cisco IOS se utiliza para los ejemplos, pero todo lo que se menciona se aplica por igual a Cisco IOS-XR. Todos los grupos de multidifusión utilizados son grupos de multidifusión específica de origen (SSM).

Consulte la figura 1. Dual-Homed-Source-1. Hay dos routers PE de entrada (PE1 y PE2) y dos routers PE de salida (PE3 y PE4). El origen se encuentra en CE1 con la dirección IP 10.100.1.6.

CE1 está doblemente conectado a PE1 y PE2.

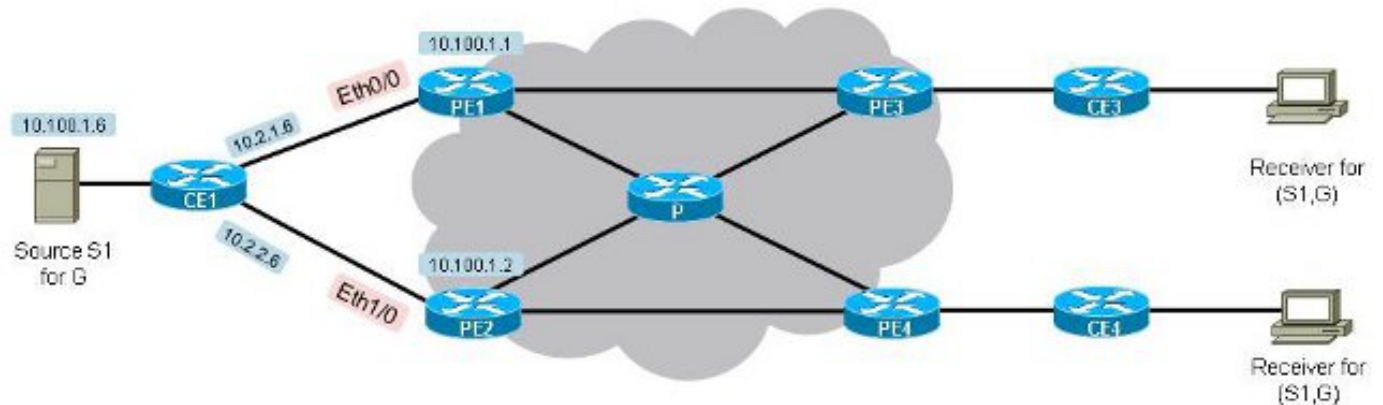


Figura 1. Fuente de doble inicio 1

La configuración en todos los routers PE (el Distinguidor de Ruta (RD) puede ser diferente en los routers PE) es:

```
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
 !
```

Para hacer que ambos routers PE de entrada comiencen a reenviar el flujo multicast (10.100.1.6.232.1.1.1) al MDT predeterminado, ambos deben recibir una Unión de un PE de egreso. Observe la topología de la Figura 1. Dual-Homed-Source-1. Puede ver que de forma predeterminada, si todos los costos de los links de borde son iguales y todos los costos de los links de núcleo son iguales, entonces PE3 hará RPF hacia PE1 y PE4 hará RPF hacia PE2 para (10.100.1.6.232.1.1.1). Ambos RPF a su PE de ingreso más cercano. Este resultado confirma lo siguiente:

```
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 tiene RPF a 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 tiene RPF a PE2. La razón por la que PE3 elige a PE1 como vecino RPF es que la ruta de unidifusión hacia 10.100.1.6/32 en Virtual Routing/Forwarding (VRF) uno es la mejor vía PE1. PE3 recibe realmente la ruta 10.100.1.6/32 tanto de PE1 como de PE2. Todos los criterios del algoritmo de cálculo de la mejor trayectoria del protocolo de gateway fronterizo (BGP) son los mismos, excepto el costo hacia la dirección del siguiente salto BGP.

```
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

```

La mejor ruta elegida por PE3 es la ruta anunciada por PE1 porque tiene el menor costo de protocolo de gateway interior (IGP) (11), frente al costo de IGP (21) hacia PE2. Para PE4 es la inversa. La topología revela que de PE3 a PE1 hay solamente un salto, mientras que de PE3 a PE2 hay dos saltos. Dado que todos los links tienen el mismo costo IGP, PE3 escoge la trayectoria de PE1 como la mejor.

La base de información de routing multidifusión (MRIB) para (10.100.1.6.232.1.1.1) se ve así en PE1 y PE2 cuando todavía no hay tráfico multidifusión:

```

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 y PE2 recibieron una conexión PIM para (10.100.1.6.232.1.1.1). La interfaz Tunnel0 se encuentra en la lista de interfaces salientes (OIL) para la entrada de multidifusión en ambos routers.

El tráfico de multidifusión comienza a fluir durante (10.100.1.6.232.1.1.1). "Debug ip pim vrf one 232.1.1.1" y "debug ip mrouting vrf one 232.1.1.1" muestran que la llegada del tráfico multicast al túnel0 (en el OIL) de ambos routers PE de entrada, hace que se ejecute el mecanismo de aserción.

## 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
```

Si la métrica y la distancia son iguales en ambos routers hacia el Origen 10.100.1.6, entonces hay un desempate para determinar el ganador de aserción. El desempate es la dirección IP más alta del vecino PIM en el túnel0 (MDT predeterminado). En este caso, esto es 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 Address	Interface	Uptime/Expires	Ver	DR 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/ Mode	Nbr Count	Query Intvl	DR Prior	DR
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 elimin  el t nel0 del OIL de la entrada multicast debido a las aserciones. Dado que el PETR LEO se ha vaciado, la entrada de multidifusi n se elimina.

```
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 tiene el indicador A configurado en la interfaz Tunnel0, porque es el ganador de aserci n.

```
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 env a peri dicamente una aserci n en Tunnel0 (MDT predeterminado), justo antes de que venza el temporizador de aserci n. Como tal, PE2 sigue siendo el ganador declarado.

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]
```

## Conclusión

El mecanismo de aserción también funciona con una interfaz de túnel en el OIL. Las aserciones se intercambian sobre el MDT predeterminado cuando los routers PE de entrada reciben tráfico multicast C-(S,G) en la interfaz de túnel asociada que está en el OIL.

## Mecanismo de afirmación con MDT de datos

La mayor parte del tiempo en que se configuran MDT de datos, el mecanismo de aserción seguirá ejecutándose en el MDT predeterminado, ya que el tráfico C-(S,G) sólo se conmuta de MDT predeterminado a los MDT de datos después de tres segundos. Luego ocurre lo mismo que se describió anteriormente. Tenga en cuenta que **sólo hay una interfaz de túnel por VRF habilitado para multicast**: el MDT predeterminado y todos los MDT de datos utilizan solamente una interfaz de túnel. Esta interfaz de túnel se utiliza en el OIL en los routers PE de entrada o como una interfaz RPF en los routers PE de salida.

En algunos casos es posible que el mecanismo de aserción no se active antes de que se señalicen los MDT de datos. A continuación, es posible que el tráfico de multidifusión C-(S,G) comience a reenviarse en un MDT de datos en ambos routers PE de entrada PE1 y PE2. En estos casos, esto podría conducir a un tráfico multicast C-(S,G) duplicado permanente a través de la red de núcleo MPLS. Para evitar esto, se implementó esta solución: cuando un router PE de entrada ve otro router PE de entrada anunciar un MDT de datos para el cual el router PE también es un router PE de entrada, se une a ese MDT de datos. En principio, sólo los routers PE de salida (que tienen un receptor descendente) se unirían al MDT de datos. Debido a que los routers PE de entrada se unen a la MDT de datos anunciada por otros routers PE de entrada, lleva al router PE de entrada que recibe tráfico multicast de la interfaz de túnel que está presente en el OIL, y por lo tanto esto activa el mecanismo de aserción y lleva a uno de los routers PE de entrada para detener el reenvío del tráfico multicast C-(S,G) a su MDT de datos (con la interfaz de túnel), mientras que el otro PE de entrada (el ganador de aserr) continúe reenviando el tráfico multidifusión C-(S,G) a su Data MDT.

Para el siguiente ejemplo, supongamos que los routers PE de entrada PE1 y PE2 nunca vieron el tráfico multicast C-(S,G) entre sí en el MDT predeterminado. El tráfico se encuentra en el MDT predeterminado durante sólo tres segundos y no es difícil entender que esto puede ocurrir si, por ejemplo, se produce una pérdida de tráfico temporal en la red principal.

La configuración para Data MDT se agrega a todos los routers PE. La configuración en todos los routers PE (el RD puede ser diferente en los routers PE) es:

```
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
 !
```

Tan pronto como PE1 y PE2 vean el tráfico del Origen, crean una entrada C-(S,G). Ambos routers PE de entrada reenvían el tráfico multicast C-(S,G) al MDT predeterminado. Los routers PE de salida PE3 y PE4 reciben el tráfico multicast y lo reenvían. Debido a un problema temporal, PE2 no ve el tráfico de PE1 y viceversa en el MDT predeterminado. Ambos envían un valor TLV (del inglés Data MDT Join Type Length Value, valor de longitud de tipo de unión de MDT) en el MDT predeterminado.

Si no hay tráfico C-(S,G), verá este estado multicast en los routers PE de entrada:

```
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
```

El indicador Y todavía no está configurado. Ambos routers PE de ingreso tienen la interfaz Tunnel0 en el OIL. Esto se debe al hecho de que PE3 tiene RPF hacia PE1 y PE4 tiene RPF hacia PE2 para C-(S,G).

Cuando el tráfico multicast para C-(S,G) comienza a fluir, tanto PE1 como PE2 reenvían el tráfico.



El umbral para MDT de datos se cruza en ambos routers PE de entrada y ambos envían un TLV de unión de MDT de datos y después de tres segundos comienzan a reenviar a su MDT de datos. Observe que PE1 se une al MDT de datos originado por PE2 y PE2 se une al MDT de datos originado por 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
```

Tanto PE1 como PE2 reciben tráfico para C-(S,G) en la interfaz Tunnel0 (pero ahora desde el MDT de datos, no el MDT predeterminado) y el mecanismo de aserción se inicia. Sólo PE2 continúa reenviando el tráfico C-(S,G) en su 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 ya no tiene la interfaz de túnel en el 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 tiene el indicador A configurado en la interfaz Tunnel0:

```
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

## Conclusión

El mecanismo assert también funciona cuando se utilizan MDTs de datos. Las aserciones se intercambian sobre el MDT predeterminado cuando los routers PE de entrada reciben tráfico multicast C-(S,G) en la interfaz de túnel asociada que está en el OIL.