

# Verificação RPF restrita para o mVPN

## Índice

[Introdução](#)

[Informações de Apoio](#)

[Problema](#)

[Solução](#)

[Notas para o Cisco IOS](#)

[Configuração](#)

[Conclusão](#)

## Introdução

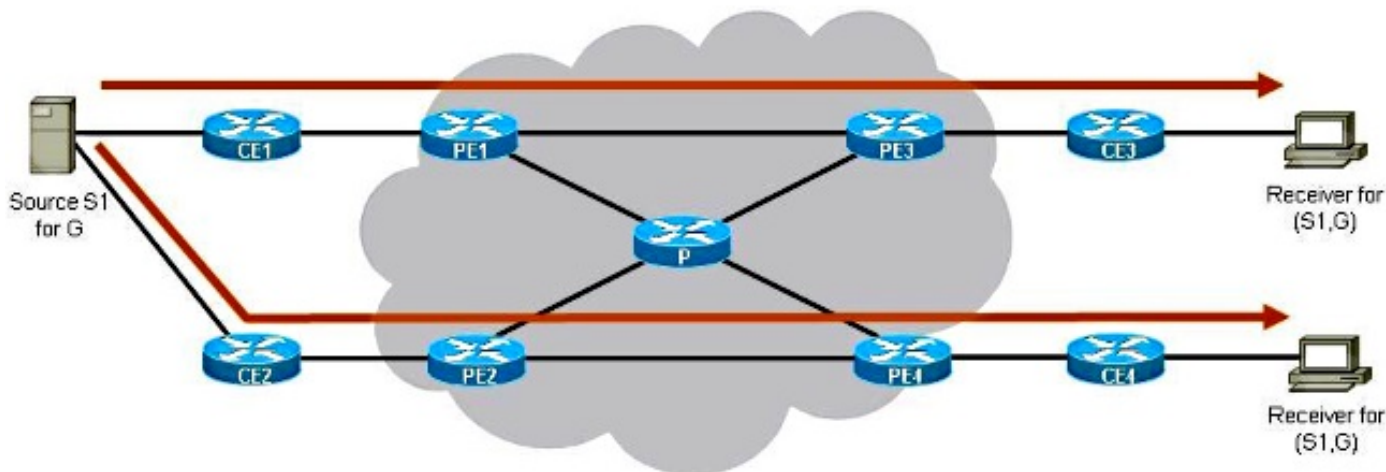
Este documento descreve a característica restrita do encaminhamento de caminho reverso (RPF) para o Multicast sobre VPN (mVPN). Este documento usa um exemplo e a aplicação no <sup>®</sup> do Cisco IOS a fim ilustrar o comportamento.

## Informações de Apoio

O RPF implica que a interface de entrada está verificada para a fonte. Embora a relação seja verificada para determinar que é correta para a fonte, não se verifica para determinar que é o vizinho de RPF correto nessa relação. Em uma relação do multi-acesso, poderia haver mais de um vizinho a que você poderia RPF. O resultado poderia ser que o roteador recebe duas vezes o mesmo fluxo de transmissão múltipla nessa relação e para a frente em ambos.

Nas redes aonde a transmissão múltipla independente de protocolo (PIM) é executado na relação do multi-acesso, esta não é uma edição, porque o fluxo de transmissão múltipla duplicado faz com que o mecanismo da afirmação seja executado e um fluxo de transmissão múltipla será recebido já não. Em alguns casos, o PIM não é executado na árvore de distribuição do Multicast (MDT), que é uma relação do multi-acesso. Nos casos, o Border Gateway Protocol (BGP) é o protocolo de sinalização da folha de prova.

Nos perfis com MDT dividido, mesmo se o PIM é executado como o protocolo da folha de prova, pode ser impossível ter afirma. A razão para esta é que uma ponta de provedor do ingresso (PE) não se junta ao MDT dividido de um outro ingresso PE nas encenações onde há dois ou mais roteadores de PE do ingresso. Cada roteador de PE do ingresso pode enviar o fluxo de transmissão múltipla em seu MDT dividido sem o outro roteador de PE do ingresso que vê o tráfego multicast. O fato de que dois roteadores de PE diferentes cada um da saída se juntam a um MDT para um roteador de PE diferente do ingresso para o mesmo fluxo de transmissão múltipla é um cenário válido: é chamado fonte de Anycast. Isto permite que os receptores diferentes juntem-se ao mesmo fluxo de transmissão múltipla mas sobre um trajeto diferente no núcleo do Multiprotocol Label Switching (MPLS). Veja figura 1 para um exemplo da fonte de Anycast.



**Figura 1**

Há dois roteadores de PE do ingresso: PE1 e PE2. Há dois roteadores de PE da saída: PE3 e PE4. Cada roteador de PE da saída tem um roteador de PE diferente do ingresso como seu vizinho de RPF. PE3 tem o PE1 como seu vizinho de RPF. PE4 tem o PE2 como seu vizinho de RPF. Os roteadores de PE da saída escolhem seu roteador de PE mais próximo do ingresso como seu vizinho de RPF.

O córrego (S1,G) irá do S1 ao receptor 1 sobre o trajeto superior e do S1 ao receptor 2 sobre o trajeto inferior. Não há nenhuma interseção dos dois córregos sobre os dois trajetos (cada trajeto no núcleo MPLS é um MDT dividido diferente).

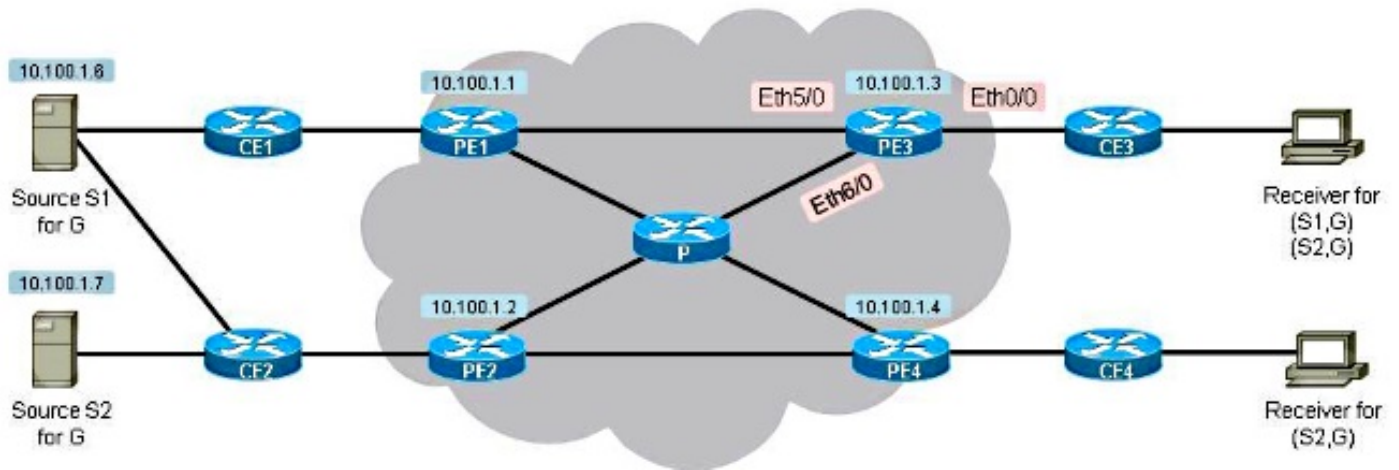
Se o MDT era um padrão MDT - tal como dentro dentro os perfis do padrão MDT - então este não trabalharia porque os dois fluxos de transmissão múltipla estariam no mesmo padrão MDT e o mecanismo da afirmação seria executado. Se o MDT é uns dados MDT nos perfis do padrão MDT, a seguir todos os roteadores de PE do ingresso juntam-se aos dados MDT dos outros roteadores de PE do ingresso e como tais veem o tráfego multicast de se e do mecanismo da afirmação são executado outra vez. Se o protocolo da folha de prova é BGP, a seguir há uma seleção ascendente do salto do Multicast (UMH) e somente um roteador de PE do ingresso é selecionado como o remetente, mas este é pelo MDT.

A fonte de Anycast é uma das vantagens grandes de executar o MDT dividido.

## Problema

A verificação RPF regular confirma que os pacotes chegam no roteador da relação correta RPF. Não há nenhuma verificação para confirmar que os pacotes estão recebidos do vizinho de RPF correto nessa relação.

Veja figura 2. Mostra a uma edição aonde o tráfego da duplicata é enviado persistentemente em uma encenação com MDT dividido. Mostra que a verificação RPF regular no caso do MDT dividido não é suficiente a fim evitar o tráfego duplicado.



**Figura 2**

Há dois receptores. O primeiro receptor estabelece-se para receber o tráfego para (S1,G) e (S2,G). O segundo receptor estabelece-se para receber o tráfego para (S2,G) somente. Há MDT dividido, e o BGP é o protocolo de sinalização da folha de prova. Note que a fonte S1 é alcançável através do PE1 e do PE2. O protocolo da núcleo-árvore é protocolo multiponto da distribuição de rótulo (mLDP).

Cada roteador de PE anuncia um tipo-1 rota do mVPN do IPv4 BGP, que indique que é um candidato a ser a raiz de um MDT dividido.

```
PE3#show bgp ipv4 mvpn vrf one
BGP table version is 257, local router ID is 10.100.1.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-pah, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:3 (default for vrf one)					
*>i [1][1:3][10.100.1.1]/12	10.100.1.1	0	100	0	?
*>i [1][1:3][10.100.1.2]/12	10.100.1.2	0	100	0	?
*> [1][1:3][10.100.1.3]/12	0.0.0.0			32768	?
*>i [1][1:3][10.100.1.4]/12	10.100.1.4	0	100	0	?

PE3 encontra o PE1 como o vizinho de RPF para o S1 depois que uma consulta para a rota do unicast para o S1.

```
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 16
Paths: (2 available, best #2, table one)
Advertised to update-groups:
 5
Refresh Epoch 2
65001, 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 0, localpref 100, valid, internal
   Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
```

```
Originator: 10.100.1.2, Cluster list: 10.100.1.5
mpls labels in/out nolabel/20
rx pathid: 0, tx pathid: 0
Refresh Epoch 2
65001, 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 0, localpref 100, valid, internal, best
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.1:1
Originator: 10.100.1.1, Cluster list: 10.100.1.5
mpls labels in/out nolabel/29
rx pathid: 0, tx pathid: 0x0
```

```
PE3#show ip rpf vrf one 10.100.1.6
```

```
RPF information for ? (10.100.1.6)
```

```
RPF interface: Lspvif0
```

```
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
```

```
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE3 seleciona o PE1 como o vizinho de RPF para (S1,G) e junta-se ao MDT dividido com o PE1 como a raiz. PE3 seleciona o PE2 como o vizinho de RPF para (S2,G) e junta-se ao MDT dividido com o PE2 como a raiz.

```
PE3#show bgp vpnv4 unicast vrf one 10.100.1.7/32
```

```
BGP routing table entry for 1:3:10.100.1.7/32, version 18
```

```
Paths: (1 available, best #1, table one)
```

```
Advertised to update-groups:
```

```
6
```

```
Refresh Epoch 2
```

```
65002, imported path from 1:2:10.100.1.7/32 (global)
```

```
10.100.1.2 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
```

```
Origin incomplete, metric 0, localpref 100, valid, internal, best
```

```
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
```

```
Originator: 10.100.1.2, Cluster list: 10.100.1.5
```

```
mpls labels in/out nolabel/29
```

```
rx pathid: 0, tx pathid: 0x0
```

```
PE3#show ip rpf vrf one 10.100.1.7
```

```
RPF information for ? (10.100.1.7)
```

```
RPF interface: Lspvif0
```

```
RPF neighbor: ? (10.100.1.2)
```

```
RPF route/mask: 10.100.1.7/32
```

```
RPF type: unicast (bgp 1)
```

```
Doing distance-preferred lookups across tables
```

```
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE4 seleciona o PE2 como o vizinho de RPF para (S1,G) e junta-se ao MDT dividido com o PE1 como a raiz.

```
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 138
```

```
Paths: (2 available, best #1, table one)
```

```
Advertised to update-groups:
```

```
2
```

```
Refresh Epoch 2
```

```
65001, 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 0, localpref 100, valid, internal, best
```

```
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
```

```

Originator: 10.100.1.2, Cluster list: 10.100.1.5
mpls labels in/out nolabel/20
rx pathid: 0, tx pathid: 0x0
Refresh Epoch 2
65001, 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 0, localpref 100, valid, internal
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.1:1
Originator: 10.100.1.1, Cluster list: 10.100.1.5
mpls labels in/out nolabel/29
rx pathid: 0, tx pathid: 0

```

PE4#show ip rpf vrf one 10.100.1.6

RPF information for ? (10.100.1.6)

**RPF interface: Lspvif0**

**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

RPF topology: ipv4 multicast base, originated from ipv4 unicast base

Observe que a relação RPF é Lspvif0 para S1 (10.100.1.6) e S2 (10.100.1.7).

PE3 junta-se ao MDT dividido do PE2 para (S2,G), e PE4 junta-se ao MDT dividido do PE2 para (S1,G). O PE1 junta-se ao MDT dividido do PE1 para (S1,G). Você pode ver este pelo tipo rotas do mVPN do IPv4 7 BGP recebidas no PE1 e no PE2.

PE1#show bgp ipv4 mvpn vrf one

BGP table version is 302, local router ID is 10.100.1.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,  
r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,  
x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:1 (default for vrf one)					
*>i [7][1:1][1][10.100.1.6/32][232.1.1.1/32]/22	10.100.1.3	0	100	0	?

PE2#show bgp ipv4 mvpn vrf one

BGP table version is 329, local router ID is 10.100.1.2

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,  
r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,  
x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:2 (default for vrf one)					
*>i [7][1:2][1][10.100.1.6/32][232.1.1.1/32]/22	10.100.1.4	0	100	0	?
*>i [7][1:2][1][10.100.1.7/32][232.1.1.1/32]/22	10.100.1.3	0	100	0	?

As entradas de transmissão múltipla em PE3 e em PE4:

PE3#show ip mroute vrf one 232.1.1.1

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

(10.100.1.7, 232.1.1.1), 21:18:24/00:02:46, flags: sTg

Incoming interface: Lspvif0, **RPF nbr 10.100.1.2**

Outgoing interface list:

Ethernet0/0, Forward/Sparse, 00:11:48/00:02:46

(10.100.1.6, 232.1.1.1), 21:18:27/00:03:17, flags: sTg

Incoming interface: Lspvif0, **RPF nbr 10.100.1.1**

Outgoing interface list:

Ethernet0/0, Forward/Sparse, 00:11:48/00:03:17

PE4#**show ip mroute vrf one 232.1.1.1**

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), 20:50:13/00:02:37, flags: sTg

Incoming interface: Lspvif0, **RPF nbr 10.100.1.2**

Outgoing interface list:

Ethernet0/0, Forward/Sparse, 20:50:13/00:02:37

Isto mostra que PE3 se junta à árvore (P2MP) point-to-multipoint enraizada no PE1 e igualmente à árvore enraizada no PE2:

PE3#**show mpls mldp database**

\* Indicates MLDP recursive forwarding is enabled

LSM ID : A Type: P2MP Uptime : 00:18:40

**FEC Root : 10.100.1.1**

Opaque decoded : [gid 65536 (0x00010000)]

Opaque length : 4 bytes

Opaque value : 01 0004 00010000

Upstream client(s) :

10.100.1.1:0 [Active]

Expires : Never Path Set ID : A

Out Label (U) : None Interface : Ethernet5/0\*

Local Label (D): 29 Next Hop : 10.1.5.1

Replication client(s):

```
MDT (VRF one)
  Uptime      : 00:18:40      Path Set ID : None
  Interface   : Lspvif0
```

```
LSM ID : B   Type: P2MP   Uptime : 00:18:40
```

```
FEC Root      : 10.100.1.2
```

```
Opaque decoded  : [gid 65536 (0x00010000)]
```

```
Opaque length   : 4 bytes
```

```
Opaque value    : 01 0004 00010000
```

```
Upstream client(s) :
```

```
10.100.1.5:0 [Active]
```

```
  Expires       : Never           Path Set ID : B
```

```
  Out Label (U) : None           Interface   : Ethernet6/0*
```

```
  Local Label (D): 30           Next Hop    : 10.1.3.5
```

```
Replication client(s):
```

```
MDT (VRF one)
```

```
  Uptime      : 00:18:40      Path Set ID : None
```

```
  Interface   : Lspvif0
```

Isto mostra que PE4 se junta à árvore P2MP enraizada no PE2:

```
PE4#show mpls mldp database
```

```
* Indicates MLDP recursive forwarding is enabled
```

```
LSM ID : 3   Type: P2MP   Uptime : 21:17:06
```

```
FEC Root      : 10.100.1.2
```

```
Opaque decoded  : [gid 65536 (0x00010000)]
```

```
Opaque value    : 01 0004 00010000
```

```
Upstream client(s) :
```

```
10.100.1.2:0 [Active]
```

```
  Expires       : Never           Path Set ID : 3
```

```
  Out Label (U) : None           Interface   : Ethernet5/0*
```

```
  Local Label (D): 29           Next Hop    : 10.1.6.2
```

```
Replication client(s):
```

```
MDT (VRF one)
```

```
  Uptime      : 21:17:06      Path Set ID : None
```

```
  Interface   : Lspvif0
```

O S1 e o S2 fluem para o grupo 232.1.1.1 com 10 pps. Você pode ver os córregos em PE3 e em PE4. Contudo, em PE3, você pode ver a taxa para (S1,G) como 20 pps.

```
PE3#show ip mroute vrf one 232.1.1.1 count
```

```
Use "show ip mfib count" to get better response time for a large number of mroutes.
```

```
IP Multicast Statistics
```

```
3 routes using 1692 bytes of memory
```

```
2 groups, 1.00 average sources per group
```

```
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
```

```
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)
```

```
Group: 232.1.1.1, Source count: 2, Packets forwarded: 1399687, Packets received: 2071455
```

```
Source: 10.100.1.7/32, Forwarding: 691517/10/28/2, Other: 691517/0/0
```

```
Source: 10.100.1.6/32, Forwarding: 708170/20/28/4, Other: 1379938/671768/0
```

```
PE4#show ip mroute vrf one 232.1.1.1 count
```

```
Use "show ip mfib count" to get better response time for a large number of mroutes.
```

```
IP Multicast Statistics
```

```
2 routes using 1246 bytes of memory
```

```
2 groups, 0.50 average sources per group
```

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second  
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.1.1.1, Source count: 1, Packets forwarded: 688820, Packets received: 688820

Source: 10.100.1.6/32, Forwarding: 688820/10/28/2, Other: 688820/0/0

```
PE3#show interfaces ethernet0/0 | include rate
Queueing strategy: fifo
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 9000 bits/sec, 30 packets/sec
```

Há um córrego duplicado. Esta duplicação é o resultado da presença do córrego (S1,G) no MDT dividido do PE1 e no MDT dividido do PE2. Este MDT dividido segundo, do PE2, foi juntado por PE3 a fim obter o córrego (S2,G). Mas, porque PE4 se juntou ao MDT dividido do PE2 a fim obter (S1,G), (S1,G) está igualmente atual no MDT dividido do PE2. Daqui, PE3 recebe o córrego (S1,G) de ambos os MDT que divididos se juntou.

PE3 não pode discriminar entre os pacotes para (S1,G) ele recebe do PE1 e do PE2. Ambos os córregos são recebidos na relação correta RPF: Lspvif0.

```
PE3#show ip multicast vrf one mpls vif
```

Interface	Next-hop	Application	Ref-Count	Table / VRF name	Flags
Lspvif0	0.0.0.0	MDT	N/A	1 (vrf one) 0x1	

Os pacotes podiam chegar em interfaces física entrantes diferentes em PE3 ou na mesma relação. Em todo caso, os pacotes dos córregos diferentes para (S1,G) chegam com uma etiqueta diferente MPLS em PE3:

```
PE3#show mpls forwarding-table vrf one
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
29	[T] No Label	[gid 65536 (0x00010000)][V]	768684	\	aggregate/one	
30	[T] No Label	[gid 65536 (0x00010000)][V]	1535940	\	aggregate/one	

[T] Forwarding through a LSP tunnel.  
View additional labelling info with the 'detail' option

## Solução

A solução é ter um RPF mais restrito. Com RPF restrito, o roteador verifica de que vizinho os pacotes são recebidos na relação RPF. Sem RPF restrito, a única verificação é determinar se a interface de entrada é a relação RPF, mas não se os pacotes estão recebidos do vizinho de RPF correto nessa relação.

## Notas para o Cisco IOS

Estão aqui algumas observações importantes sobre o RPF com Cisco IOS.

- Quando você muda para/desde o modo restrito RPF, qualquer um configurar-lo antes que você configure o MDT dividido ou o BGP claro. Se você configura somente o comando RPF



restrito, não criará uma outra relação de Lspvif imediatamente.

- O RPF restrito não é permitido à revelia no Cisco IOS.
- Não é apoiado para ter o comando restrito-RPF com perfis do padrão MDT.

## Configuração

Você pode configurar o RPF restrito em PE3 para o roteamento virtual e a transmissão (VRF).

```
vrf definition one
rd 1:3
!
address-family ipv4
mdt auto-discovery mldp
  mdt strict-rpf interface
  mdt partitioned mldp p2mp
mdt overlay use-bgp
route-target export 1:1
route-target import 1:1
exit-address-family
!
```

A informação de RPF mudou:

```
PE3#show ip rpf vrf one 10.100.1.6
RPF information for ? (10.100.1.6)
  RPF interface: Lspvif0
Strict-RPF interface: Lspvif1
  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
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

```
PE3#show ip rpf vrf one 10.100.1.7
RPF information for ? (10.100.1.7)
  RPF interface: Lspvif0
Strict-RPF interface: Lspvif2
  RPF neighbor: ? (10.100.1.2)
RPF route/mask: 10.100.1.7/32
RPF type: unicast (bgp 1)
Doing distance-preferred lookups across tables
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE3 criou uma relação de Lspvif pelo ingresso PE. A relação de Lspvif é criada pelo ingresso PE, pela família do endereço (AF), e pelo VRF. O RPF para 10.100.1.6 aponta agora para conectar Lspvif1 e o RPF para 10.100.1.7 aponta agora para conectar Lspvif2.

```
PE3#show ip multicast vrf one mpls vif
```

Interface	Next-hop	Application	Ref-Count	Table / VRF name	Flags
Lspvif0	0.0.0.0	MDT	N/A	1 (vrf one)	0x1
Lspvif1	10.100.1.1	MDT	N/A	1 (vrf one)	0x1
Lspvif2	10.100.1.2	MDT	N/A	1 (vrf one)	0x1

Agora, a verificação RPF para os pacotes (S1,G) do PE1 é verificada contra a relação Lspvif1 RPF. Estes pacotes entram com etiqueta 29 MPLS. A verificação RPF para os pacotes (S2,G) do

PE2 é verificada contra a relação Lspvif2 RPF. Estes pacotes entram com etiqueta 30 MPLS. Os córregos chegam em PE3 através das interfaces de entrada diferentes, mas esta poderia igualmente ser a mesma relação. Contudo, devido ao fato de que o mLDP nunca usa o Penultimate Hop Popping (PHP), há sempre uma etiqueta regular MPLS sobre os pacotes de transmissão múltipla. Os pacotes (S1,G) que chegam do PE1 e do PE2 estão em dois MDT divididos diferentes e têm consequentemente uma etiqueta diferente MPLS. Daqui, PE3 pode discriminar entre o córrego (S1,G) que vem do PE1 e do córrego (S1,G) que vem do PE2. Desta maneira, os pacotes podem ser mantidos separados por PE3 e um RPF pode ser executado contra roteadores de PE diferentes do ingresso.

O base de dados do mLDP em PE3 mostra agora as relações diferentes de Lspvif pelo ingresso PE.

```
PE3#show mpls mldp database
```

```
* Indicates MLDP recursive forwarding is enabled
```

```
LSM ID : C   Type: P2MP   Uptime : 00:05:58
FEC Root      : 10.100.1.1
Opaque decoded : [gid 65536 (0x00010000)]
Opaque length  : 4 bytes
Opaque value   : 01 0004 00010000
Upstream client(s) :
  10.100.1.1:0   [Active]
    Expires      : Never           Path Set ID : C
    Out Label (U) : None           Interface   : Ethernet5/0*
    Local Label (D): 29           Next Hop    : 10.1.5.1
Replication client(s):
  MDT (VRF one)
    Uptime       : 00:05:58       Path Set ID : None
    Interface    : Lspvif1
```

```
LSM ID : D   Type: P2MP   Uptime : 00:05:58
FEC Root      : 10.100.1.2
Opaque decoded : [gid 65536 (0x00010000)]
Opaque length  : 4 bytes
Opaque value   : 01 0004 00010000
Upstream client(s) :
  10.100.1.5:0   [Active]
    Expires      : Never           Path Set ID : D
    Out Label (U) : None           Interface   : Ethernet6/0*
    Local Label (D): 30           Next Hop    : 10.1.3.5
Replication client(s):
  MDT (VRF one)
    Uptime       : 00:05:58       Path Set ID : None
    Interface    : Lspvif2
```

RPF restrito ou RPF pelos trabalhos do ingresso PE devido ao fato de que os fluxos de transmissão múltipla entram ao ingresso PE com uma etiqueta diferente MPLS pelo ingresso PE:

```
PE3#show mpls forwarding-table vrf one
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
29	[T] No Label	[gid 65536 (0x00010000)][V]	162708	\	aggregate/one	
30	[T] No Label	[gid 65536 (0x00010000)][V]	162750	\	aggregate/one	

```
[T] Forwarding through a LSP tunnel.
View additional labelling info with the 'detail' option
```

A prova que o RPF restrito trabalha é que há já não um córrego duplicado (S1,G) enviado em PE3. O córrego duplicado ainda chega em PE3, mas é deixado cair devido à falha de RPF. O contador da falha de RPF está em 676255 e em aumentos constantemente a uma taxa de 10 pps.

```
PE3#show ip mroute vrf one 232.1.1.1 count
```

Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics

3 routes using 1692 bytes of memory

2 groups, 1.00 average sources per group

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second

Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.1.1.1, Source count: 2, Packets forwarded: 1443260, Packets received: 2119515

Source: 10.100.1.7/32, Forwarding: 707523/10/28/2, Other: 707523/0/0

Source: 10.100.1.6/32, Forwarding: 735737/10/28/2, Other: 1411992/676255/0

A taxa de emissor em PE3 é agora 20 pps, que é 10 pps para cada córrego (S1,G) e (S2,G):

```
PE3#show interfaces ethernet0/0 | include rate
```

Queueing strategy: fifo

30 second input rate 0 bits/sec, 0 packets/sec

30 second output rate 6000 bits/sec, 20 packets/sec

## Conclusão

A verificação RPF restrita deve ser usada para os modelos de distribuição do mVPN que usam o MDT dividido.

As coisas puderam parecer trabalhar, mesmo se você não configura a verificação RPF restrita para os modelos de distribuição do mVPN com MDT dividido: os fluxos de transmissão múltipla são entregados aos receptores. Contudo, há a possibilidade de que há um tráfego multicast duplicado quando as fontes são conectadas aos roteadores de PE múltiplos do ingresso. Isto conduz a um desperdício da largura de banda na rede e pode adversamente afetar o aplicativo multicast nos receptores. Daqui, é uma obrigação para configurar a verificação RPF restrita para os modelos de distribuição do mVPN que os usos dividiram o MDT.