

Verificação RPF RPF Rígida para mVPN

Contents

[Introduction](#)

[Informações de Apoio](#)

[Problema](#)

[Solução](#)

[Notas para o Cisco IOS](#)

[Configuração](#)

[Conclusão](#)

Introduction

Este documento descreve o recurso RPF (Strict Reverse Path Forwarding) para Multicast sobre VPN (mVPN). Este documento usa um exemplo e a implementação no Cisco IOS[®] para ilustrar o comportamento.

Informações de Apoio

O RPF implica que a interface de entrada é verificada em direção à origem. Embora a interface seja verificada para determinar se é a correta na direção da origem, ela não é verificada para determinar se é o vizinho RPF correto nessa interface. Em uma interface multiacesso, pode haver mais de um vizinho para o qual você poderia executar RPF. O resultado pode ser que o roteador receba duas vezes o mesmo fluxo multicast naquela interface e encaminhe ambas.

Em redes onde o Protocol Independent Multicast (PIM) é executado na interface multiacesso, isso não é um problema, porque o fluxo multicast duplicado faz com que o mecanismo assert seja executado e um fluxo multicast não será mais recebido. Em alguns casos, o PIM não é executado na árvore de distribuição multicast (MDT), que é uma interface multiacesso. Nesses casos, o BGP (Border Gateway Protocol) é o protocolo de sinalização de sobreposição.

Nos perfis com MDT particionado, mesmo que o PIM seja executado como o protocolo de sobreposição, pode ser impossível ter asserções. Isso ocorre porque uma Borda do provedor de entrada (PE) não se junta ao MDT particionado de outro PE de entrada nos cenários em que há dois ou mais roteadores PE de entrada. Cada roteador de PE de entrada pode encaminhar o fluxo multicast para seu MDT particionado sem que o outro roteador de PE de entrada veja o tráfego multicast. O fato de que dois roteadores PE de saída diferentes unem um MDT para um roteador PE de entrada diferente para o mesmo fluxo multicast é um cenário válido: é chamado de Origem de Anycast. Isso permite que diferentes receptores se juntem ao mesmo fluxo multicast, mas em um caminho diferente no núcleo de Multiprotocol Label Switching (MPLS). Veja na Figura 1 um exemplo de Origem de Anycast.

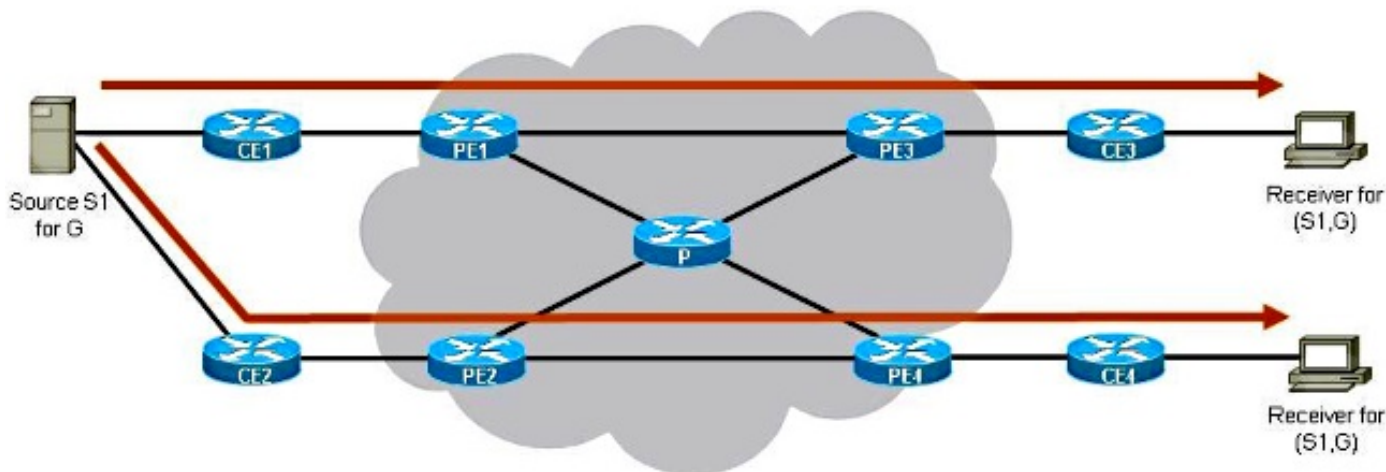


Figure 1

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

O fluxo (S1,G) vai do S1 para o receptor 1 no caminho superior e do S1 para o receptor 2 no caminho inferior. Não há interseção dos dois fluxos sobre os dois caminhos (cada caminho no núcleo MPLS é um MDT particionado diferente).

Se o MDT fosse um MDT padrão - como nos perfis de MDT padrão - isso não funcionaria porque os dois fluxos multicast estariam no mesmo MDT padrão e o mecanismo de asserção seria executado. Se o MDT for um MDT de dados nos perfis de MDT padrão, todos os roteadores de PE de entrada se juntarão ao MDT de dados dos outros roteadores de PE de entrada e, como tal, verão o tráfego multicast um do outro e o mecanismo de asserção será executado novamente. Se o protocolo de sobreposição for BGP, então há a seleção Upstream Multicast Hop (UMH) e somente um roteador Ingress PE é selecionado como o encaminhador, mas isso é por MDT.

Anycast Source é uma das grandes vantagens da execução do MDT particionado.

Problema

A verificação RPF regular confirma que os pacotes chegam ao roteador a partir da interface RPF correta. Não há verificação para confirmar se os pacotes são recebidos do vizinho RPF correto nessa interface.

Consulte a Figura 2. Ele mostra um problema em que o tráfego duplicado é encaminhado persistentemente em um cenário com MDT particionado. Mostra que a verificação RPF regular no caso de MDT particionado não é suficiente para evitar tráfego duplicado.

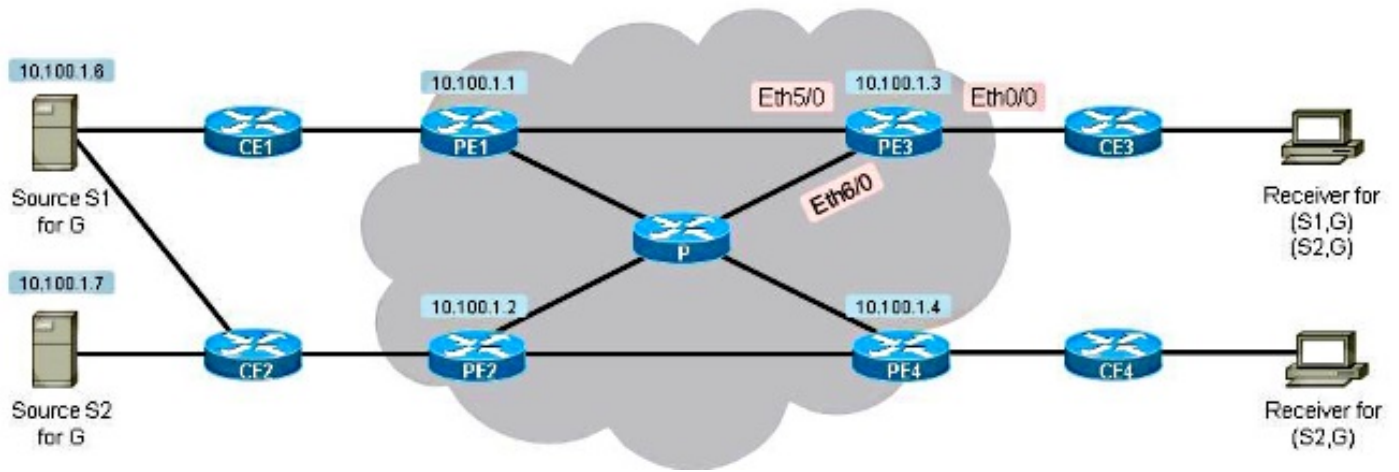


Figure 2

Há dois receptores. O primeiro receptor é configurado para receber tráfego para (S1,G) e (S2,G). O segundo receptor é configurado para receber tráfego somente para (S2,G). Há o MDT particionado e o BGP é o protocolo de sinalização de sobreposição. Observe que a origem S1 pode ser alcançada via PE1 e PE2. O protocolo da árvore central é o Multipoint Label Distribution Protocol (mLDP).

Cada roteador PE anuncia uma rota mVPN IPv4 BGP tipo 1, que indica que é um candidato a ser a raiz de um MDT particionado.

```
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 PE1 como vizinho RPF para S1 após uma pesquisa da rota unicast para 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 PE1 como o vizinho RPF para (S1,G) e une o MDT Particionado com PE1 como raiz. PE3 seleciona PE2 como o vizinho RPF para (S2,G) e une o MDT Particionado com PE2 como 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 PE2 como o vizinho RPF para (S1,G) e une o MDT Particionado com PE1 como 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 interface RPF é Lspvif0 para S1 (10.100.1.6) e S2 (10.100.1.7).

PE3 une o MDT particionado de PE2 para (S2,G) e PE4 ao MDT particionado de PE2 para (S1,G). PE1 une o MDT particionado de PE1 para (S1,G). Você pode ver isso pelas rotas mVPN IPv4 BGP tipo 7 recebidas em PE1 e 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 multicast em PE3 e 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

Isso mostra que PE3 une a árvore ponto a multiponto (P2MP) com raiz em PE1 e também a árvore com raiz em 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

```

Isso mostra que PE4 se une à árvore P2MP com raiz em 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

```

Fluxo S1 e S2 para o Grupo 232.1.1.1 com 10 pps. Você pode ver os fluxos em PE3 e PE4. No entanto, 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 fluxo duplicado. Esta duplicação é o resultado da presença de fluxo (S1,G) no MDT Particionado de PE1 e no MDT Particionado de PE2. Este segundo MDT Particionado, do PE2, foi unido por PE3 para obter o fluxo (S2,G). Mas, porque PE4 se juntou ao MDT Particionado de PE2 para obter (S1,G), (S1,G) também está presente no MDT Particionado de PE2. Portanto, o PE3 recebe o fluxo (S1,G) de ambos os MDTs particionados em que se juntou.

PE3 não pode discriminar os pacotes para (S1,G) que recebe de PE1 e PE2. Ambos os fluxos são recebidos na interface RPF correta: 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 podem chegar em diferentes interfaces físicas de entrada em PE3 ou na mesma interface. Em qualquer caso, os pacotes dos diferentes fluxos para (S1,G) chegam com um rótulo MPLS diferente 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 rigoroso. Com RPF estrito, o roteador verifica de que vizinho os pacotes são recebidos na interface RPF. Sem RPF rigoroso, a única verificação é determinar se a interface de entrada é a interface RPF, mas não se os pacotes são recebidos do vizinho RPF correto nessa interface.

Notas para o Cisco IOS

Aqui estão algumas notas importantes sobre RPF com Cisco IOS.

- Quando você altera para/do modo RPF estrito, configure-o antes de configurar o MDT particionado ou limpe o BGP. Se você configurar apenas o comando RPF strict, ele não criará outra interface Lspvif imediatamente.

- RPF restrito não está habilitado por padrão no Cisco IOS.
- Não é suportado ter o comando **strict-rpf** com perfis de MDT padrão.

Configuração

Você pode configurar RPF estrito em PE3 para o Virtual Routing and Forwarding (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
!
```

As informações de RPF foram alteradas:

```
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
```

O PE3 criou uma interface Lspvif por PE de entrada. A interface Lspvif é criada por PE de entrada, por família de endereços (AF) e por VRF. O RPF para 10.100.1.6 agora aponta para a interface Lspvif1 e o RPF para 10.100.1.7 agora aponta para a interface 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 de RPF para pacotes (S1,G) de PE1 é verificada na interface Lspvif1 de RPF. Esses pacotes vêm com o rótulo 29 do MPLS. A verificação de RPF para pacotes (S2,G) do PE2 é verificada em relação à interface Lspvif2 do RPF. Esses pacotes entram com o rótulo MPLS 30. Os fluxos chegam ao PE3 através de diferentes interfaces de entrada, mas essa

também pode ser a mesma interface. No entanto, devido ao fato de que o mLDP nunca usa Penúltimos-saltos-saltos (PHP), sempre há um rótulo MPLS regular em cima dos pacotes multicast. Os pacotes (S1,G) que chegam de PE1 e PE2 estão em dois MDTs particionados diferentes e, portanto, têm um rótulo MPLS diferente. Assim, o PE3 pode diferenciar entre o fluxo (S1,G) que vem do PE1 e o fluxo (S1,G) que vem do PE2. Dessa forma, os pacotes podem ser mantidos separados por PE3 e um RPF pode ser executado em diferentes roteadores PE de entrada.

O banco de dados mLDP em PE3 agora mostra as diferentes interfaces Lspvif por PE de entrada.

```
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 ou RPF por PE de entrada funciona devido ao fato de que os fluxos multicast chegam ao PE de entrada com um rótulo MPLS diferente por PE de entrada:

```
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 de que o RPF rigoroso funciona é que não há mais um fluxo duplicado (S1,G) encaminhado no PE3. O fluxo duplicado ainda chega em PE3, mas é descartado devido à falha de RPF. O contador de falha de RPF está em 676255 e aumenta 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 saída em PE3 agora é de 20 pps, que é de 10 pps para cada fluxo (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 estrita deve ser usada para os modelos de implantação mVPN que usam MDT particionado.

As coisas podem parecer funcionar, mesmo que você não configure a verificação RPF estrita para os modelos de implantação mVPN com MDT particionado: os fluxos multicast são entregues aos receptores. No entanto, há a possibilidade de haver tráfego multicast duplicado quando as fontes estão conectadas a vários roteadores de entrada PE. Isso leva a um desperdício de largura de banda na rede e pode afetar adversamente a aplicação multicast nos receptores. Portanto, é necessário configurar a Verificação RPF estrita para os modelos de implantação de mVPN que usam MDT particionado.