

Строгая Проверка переадресации по обратному пути для MVPN

Содержание

[Введение](#)

[Общие сведения](#)

[Проблема](#)

[Решение](#)

[Примечания для Cisco IOS](#)

[!--- конфигурацию](#)

[Заключение](#)

Введение

Этот документ описывает Строгую функцию Пересылки по обратному пути (RPF) Групповой адресации по VPN (MVPN). Этот документ использует пример и реализацию в Cisco IOS® для иллюстрирования поведения.

Общие сведения

RPF подразумевает, что входящий интерфейс проверен к источнику. Несмотря на то, что интерфейс проверен, чтобы решить, что это - корректное к источнику, это не проверено, чтобы решить, что это - корректный сосед RPF на том интерфейсе. На мультиинтерфейсе доступа могло быть несколько соседних узлов, к которым вы могли RPF. Результат мог состоять в том, что маршрутизатор получает дважды ту же многоадресную рассылку на том интерфейсе и вперед обоим.

В сетях, где независимая от протокола многоадресная передача (PIM) работает на мультиинтерфейсе доступа, это не проблема, потому что двойная многоадресная рассылка заставляет утверждать механизм работать, и одна многоадресная рассылка больше не будет получаться. В некоторых случаях PIM не работает на Дереве распределения групповой адресации (MDT), которое является мультиинтерфейсом доступа. В тех случаях Протокол BGP является протоколом сигнализации наложения.

В профилях с Разделенным MDT, даже если PIM выполняется как протокол наложения, может быть невозможно иметь, утверждает. Причина для этого состоит в том, что одна Входная Граница провайдера (PE) не присоединяется к Разделенному MDT от другого Входного PE в сценариях, где существует два или больше Входных Периферийных маршрутизатора. Каждый Входной Периферийный маршрутизатор может передать многоадресную рассылку на свой Разделенный MDT без другого Входного Периферийного маршрутизатора, видя многоадресный трафик. Факт, что два других Выходных Периферийных маршрутизатора каждое соединение MDT к другому Входному

Периферийному маршрутизатору для той же многоадресной рассылки являются допустимым сценарием: это называют Источником адресации любому устройству группы. Это позволяет другим приемникам присоединяться к той же многоадресной рассылке, но по другому пути в ядре Многопротокольной коммутации по меткам (MPLS). Посмотрите рисунок 1 для примера Источника адресации любому устройству группы.

Рисунок 1

Существует два Входных Периферийных маршрутизатора: PE1 и PE2. Существует два Выходных Периферийных маршрутизатора: PE3 и PE4. Каждый Выходной Периферийный маршрутизатор имеет другой Входной Периферийный маршрутизатор как своего соседа RPF. PE3 имеет PE1 как своего соседа RPF. PE4 имеет PE2 как своего соседа RPF. Выходные Периферийные маршрутизаторы выбирают свой самый близкий Входной Периферийный маршрутизатор как их соседа RPF.

Поток (S1, G) пойдет с S1 на Получатель 1 по главному пути и с S1 на Получатель 2 по нижнему пути. Нет никакого пересечения этих двух потоков по двум путям (каждым путем в ядре MPLS является другой Разделенный MDT).

Если бы MDT был По умолчанию MDT - такой как в По умолчанию профили MDT то - тогда это не работало бы, потому что эти две многоадресных рассылки будут на том же По умолчанию, который выполнили бы MDT и утверждать механизм. Если MDT является Данными MDT в По умолчанию профили MDT, то все Входные Периферийные маршрутизаторы присоединяются к Данным, MDT от других Входных Периферийных маршрутизаторов и как таковой видит многоадресный трафик друг от друга, и утверждать механизм выполняется снова. Если протокол наложения является BGP, то существует выбор Восходящего перехода групповой адресации (UMH), и только один Входной Периферийный маршрутизатор выбран как средство передачи, но это на MDT.

Источник адресации любому устройству группы является одним из больших преимуществ выполнения Разделенного MDT.

Проблема

Обычная Проверка переадресации по обратному пути подтверждает, что пакеты поступают в маршрутизатор от корректного интерфейса RPF. Нет никакой проверки, чтобы подтвердить, что пакеты получены от корректного соседа RPF на том интерфейсе.

(см. рис. 2). Это показывает проблему, где двойной трафик постоянно передается в сценарии с Разделенным MDT. Это показывает, что обычная Проверка переадресации по обратному пути в случае Разделенного MDT не достаточна во избежание двойного трафика.

Рис. 2

Существует два приемника. Первый получатель установлен для получения трафика для (S1, G) и (S2, G). Второй получатель установлен для получения трафика для (S2, G) только. Существует Разделенный MDT, и BGP является протоколом сигнализации наложения. Обратите внимание на то, что Исходный S1 достижим и через PE1 и через PE2. Базово-древовидный протокол является Многоточечным Протоколом распределения меток (mLDP).

Каждый Периферийный маршрутизатор объявляет маршрут MVPN IPv4 BGP Типа 1,

который указывает, что это - кандидат, чтобы быть root Разделенного MDT.

```
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 находит PE1 как соседа RPF для S1 после поиска для маршрута одноадресной передачи для 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 nlabel/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 nlabel/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 выбирает PE1 как соседа RPF для (S1, G) и присоединяется к Разделенному MDT с PE1 как root. PE3 выбирает PE2 как соседа RPF для (S2, G) и присоединяется к Разделенному MDT с PE2 как root.

```
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 выбирает PE2 как соседа RPF для (S1, G) и присоединяется к Разделенному MDT с PE1 как root.

```
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

Заметьте, что интерфейс RPF является Lspvif0 и для S1 (10.100.1.6) и для S2 (10.100.1.7).

PE3 присоединяется к Разделенному MDT от PE2 для (S2, G), и PE4 присоединяется к Разделенному MDT от PE2 для (S1, G). PE1 присоединяется к Разделенному MDT от PE1 для (S1, G). Вы видите это маршрутами MVPN IPv4 BGP типа 7, полученными на PE1 и 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	?

Многоадресные записи на PE3 и 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

Это показывает, что PE3 присоединяется к Точка - многоточка (P2MP), дерево базировалось в PE1, и также дерево базировалось в 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
```

Это показывает, что PE4 присоединяется, дерево P2MP базировалось в 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
```

S1 и поток S2 для Группы 232.1.1.1 с 10 pps. Вы видите потоки в PE3 и PE4. Однако в PE3, вы видите скорость для (S1, G) как 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
```

Существует двойной поток. Это дублирование является результатом присутствия потока (S1, G) на Разделенном MDT от PE1 и на Разделенном MDT от PE2. К этому второму Разделенному MDT, от PE2, присоединился PE3 для получения потока (S2, G). Но, потому что PE4 присоединился к Разделенному MDT от PE2 для получения (S1, G), (S1, G) также присутствует на Разделенном MDT от PE2. Следовательно, PE3 получает поток (S1, G) от обоих Разделенный MDTs, к которому это присоединилось.

PE3 не может различить между пакетами для (S1, G) это получает от PE1 и PE2. Оба потока получены на корректном интерфейсе 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

Пакеты могли поступить в другие входящие физические интерфейсы на PE3 или на том же интерфейсе. В любом случае пакеты от других потоков для (S1, G) действительно поступают с другим MPLS label в 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
```

Решение

Решение состоит в том, чтобы иметь более строгий RPF. Со строгим RPF маршрутизатор проверяет, от которого соседнего узла пакеты получены на интерфейсе RPF. Без строгого

RPF единственная проверка должна определить, является ли входящий интерфейс интерфейсом RPF, но не, если пакеты получены от корректного соседа RPF на том интерфейсе.

Примечания для Cisco IOS

Вот некоторые важные замечания о RPF с Cisco IOS.

- Когда вы изменяете к/от строгий режим RPF, или настраиваете его перед настройкой Разделенного MDT или clear BGP. Если вы только настроите строгую команду RPF, то она сразу не создаст другой интерфейс Lspvif.
- Строгий RPF не включен по умолчанию в Cisco IOS.
- Это не поддерживается для имени **строгой-rpf** команды с По умолчанию профили MDT.

!--- конфигурацию

Можно настроить строгий RPF на PE3 для Виртуальной маршрутизации и Передачи (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
!
```

Информация RPF изменилась:

```
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 создал интерфейс Lspvif на Входной PE. Интерфейс Lspvif создан на Входной PE на Семейство адресов (AF), и на VRF. RPF для 10.100.1.6 теперь точки для взаимодействия через интерфейс Lspvif1 и RPF для 10.100.1.7 теперь точки для взаимодействия через

интерфейс 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

Теперь, Проверка переадресации по обратному пути для пакетов (S1, G) от PE1 проверены против интерфейса RPF Lspvif1. Эти пакеты входят с MPLS label 29. Проверка переадресации по обратному пути для пакетов (S2, G) от PE2 проверены против интерфейса RPF Lspvif2. Эти пакеты входят с MPLS label 30. Потoki поступают в PE3 через другие входящие интерфейсы, но это могло также быть тем же интерфейсом. Однако вследствие того, что mLDP никогда не использует Вытеснение предпоследней пересылки (PHP), всегда существует обычный MPLS label поверх пакетов групповой адресации. (S1, G) пакеты, которые поступают от PE1 и от PE2, находятся на двух других Разделенных MDTs и поэтому имеют другого MPLS label. Следовательно, PE3 может различить между (S1, G) поток, который прибывает из PE1 и (S1, G) поток, который прибывает из PE2. Таким образом пакеты могут держаться отдельно PE3, и RPF может быть выполнен против других Входных Периферийных маршрутизаторов.

mLDP база данных по PE3 теперь показывает другие интерфейсы Lspvif на Входной 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
```

Строгий RPF или RPF на Входной PE работают вследствие того, что многоадресные рассылки входят к Входному PE с другим MPLS label на входной 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

Доказательство, что строгие работы RPF состоят в том, что больше нет двойного потока (S1, G) передан на PE3. Двойной поток все еще поступает в PE3, но это отброшено из-за Ошибки переадресации по обратному пути. Счетчик Ошибки переадресации по обратному пути в 676255 и постоянно увеличивается на скорости 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
```

Скорость передачи выходного сигнала в PE3 является теперь 20 pps, который является 10 pps для каждого потока (S1, G) и (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
```

Заключение

Строгая Проверка переадресации по обратному пути должна использоваться для моделей развертывания MVPN то использование Разделенный MDT.

Даже если вы не настраиваете строгую Проверку переадресации по обратному пути для моделей развертывания MVPN с Разделенным MDT, вещи, могло бы казаться, работали бы: многоадресные рассылки отправлены приемникам. Однако существует возможность, что существует двойной многоадресный трафик, когда источники связаны со множественными Входными Периферийными маршрутизаторами. Это приводит к трате пропускной способности в сети и может оказать негативное влияние на приложение групповой адресации на приемниках. Следовательно, это - необходимость для настройки строгой Проверки переадресации по обратному пути для моделей развертывания MVPN, которая использует Разделенный MDT.