

# mVPN での厳格な RPF チェック

## 目次

[概要](#)

[背景説明](#)

[問題](#)

[解決策](#)

[Cisco IOS についての注](#)

[設定](#)

[結論](#)

## 概要

このドキュメントでは、Multicast over VPN ( mVPN ) 用の厳密なリバースパス フォワーディング ( RPF ) 機能について説明します。このドキュメントでは、Cisco IOS<sup>®</sup> での例および実装を使用して、動作を示します。

## 背景説明

RPF では、入力インターフェイスが送信元に対してチェックされます。送信元に対する正しいインターフェイスであることを確認するためのチェックは行われますが、そのインターフェイス上の正しい RPF ネイバーであることを確認するチェックは行われません。マルチアクセスインターフェイスでは、RPF が可能なネイバーが複数ある場合があります。その場合、ルータがそのインターフェイス上で同じマルチキャスト ストリームを 2 回受信し、そのどちらも転送する可能性があります。

Protocol Independent Multicast ( PIM ) がマルチアクセス インターフェイスで動作するネットワークにおいては、マルチキャスト ストリームが重複しているとアサート機能が動作して、マルチキャスト ストリーム的一方を受信しなくなるため、問題にはなりません。場合によっては、マルチアクセス インターフェイスであるマルチキャスト 配布ツリー ( MDT ) 上で PIM が動作しません。そのような場合、ボーダー ゲートウェイ プロトコル ( BGP ) は、オーバーレイ シグナリング プロトコルです。

パーティション MDT を伴うプロファイルにおいては、PIM がオーバーレイ プロトコルとして実行されていても、アサートが不可能な場合があります。これは、複数の入力 PE ルータのあるシナリオで、1 つの入力のプロバイダー エッジ ( PE ) は別の入力 PE からのパーティション MDT に結合しないためです。各入力 PE ルータは、マルチキャスト トラフィックが表示されている他の入力 PE ルータなしで、パーティション MDT にマルチキャスト ストリームを転送できます。同じマルチキャスト ストリームについて、2 つの出力 PE ルータがそれぞれ、別の入力 PE ルータへ MDT に結合するというのは、有効なシナリオです。これはエニーキャスト ソースと呼ばれます。これにより、複数の異なる受信側が、マルチプロトコル ラベル スイッチング ( MPLS ) コア内の異なるパスで、同じマルチキャスト ストリームに結合できるようになります。エニーキャスト ソースの例については、図 1 を参照してください。

## 図 1

2つの入力 PE ルータがあります。PE1 および PE2。2つの出力 PE ルータがあります：PE3 および PE4。各出力 PE ルータには、RPF ネイバーとして別の入力 PE ルータがあります。PE3 の RPF ネイバーは PE1 です。PE4 の RPF ネイバーは PE2 です。出力 PE ルータは、RPF ネイバーとして最も近い入力 PE ルータを選択します。

S1 から Receiver 1 へのストリーム ( S1、G ) は最上位パス、S1 から Receiver 2 へは最下位パスになります。2つのパス上で2つのストリームの交差点はありません ( MPLS コアの各パスは異なるパーティション MDT です )。

MDT がデフォルト MDT ( デフォルト MDT プロファイルに含まれるものなど ) の場合、2つのマルチキャストストリームが同じデフォルト MDT 上にあり、アサートメカニズムが動作するため、これは機能しません。MDT がデフォルト MDT プロファイルのデータ MDT である場合は、すべての入力 PE ルータは他の入力 PE ルータからのデータ MDT に結合するため、互いのマルチキャストトラフィックが表示され、アサート機能が再び動作します。オーバーレイプロトコルが BGP の場合、アップストリームマルチキャストホップ ( UMH ) の選択があり、1つの入力 PE ルータのみがフォワードとして選択されますが、これは MDT によって異なります。

ユニキャストソースは、パーティション MDT を実行する大きな利点の1つです。

## 問題

定期的な RPF チェックにより、パケットが正しい RPF インターフェイスからルータに到着することが確認されます。パケットがそのインターフェイス上の正しい RPF ネイバーから受信していることを確認するチェックはありません。

図 2 を参照してください。ここには、パーティション MDT が関係するシナリオで、重複トラフィックが永続的に転送されるという問題が示されています。パーティション MDT の場合の定期的な RPF チェックでは、重複トラフィック回避のために十分ではないことが示されています。

## 図 2

次の2つのレシーバがあります。最初のレシーバはトラフィックを受信するように設定されています ( S1、G ) および ( S2、G )。2番目のレシーバは、( S2、G ) についてのみトラフィックを受信するように設定されています。パーティション MDT があり、BGP はオーバーレイシグナリングプロトコルです。ソース S1 が PE1 および PE2 の両方で到達可能であることに注意してください。コアツリープロトコルはマルチポイントラベル配布プロトコル ( mLDP ) です。

各 PE ルータは、パーティション MDT のルートの候補であることを示すタイプ 1 BGP IPv4 mVPN ルートをアドバタイズします。

```
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 は、S1 のユニキャスト ルートの照合後に、S1 の RPF ネイバーとして PE1 を検出しました。

```

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 は、( S1、G ) の RPF ネイバーとして PE1 を選択し、PE1 をルートとして持つパーティション MDT に結合します。PE3 は、( S2、G ) の RPF ネイバーとして PE2 を選択し、PE2 をルートとして持つパーティション MDT に結合します。

```

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 は、( S1、G ) の RPF ネイバーとして PE2 を選択し、PE1 をルートとして持つパーティション MDT に結合します。

```
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
```

RPF インターフェイスが、S1 ( 10.100.1.6 ) と S2 ( 10.100.1.7 ) の両方について Lspvif0 であることに注意してください。

PE3 は ( S2、G ) の PE2 からパーティション MDT に結合し、PE4 は ( S1、G ) の PE2 からパーティション MDT に結合します。PE1 は、( S1、G ) の PE1 からパーティション MDT に結合します。PE1 および PE2 で受信されたタイプ 7 BGP IPv4 mVPN ルートによりこれを確認できます。

```
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	?

PE3 および PE4 上のマルチキャスト エントリ :

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

これは、PE3 が PE1 をルートとするポイントツーマルチポイント ( P2MP ) ツリーに結合し、さらに PE2 をルートとするツリーにも結合します。

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

これは、PE4 が PE2 をルートとする P2MP ツリーに結合することを示しています :

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 と S2 は、10 pps でグループ 232.1.1.1 を流れます。PE3 と PE4 でストリームを確認できます。ただし、PE3 では、( S1、G ) の速度が 20pps となっていることが分かります。

```
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

重複ストリームがあります。この重複は、PE1 からのパーティション MDT 上と PE2 からのパーティション MDT 上にストリーム ( S1、G ) が存在することによるものです。PE2 からのこの第 2 のパーティション MDT は、ストリーム ( S2、G ) を取得するために PE3 により結合されたものです。しかし、PE4 が ( S1、G ) を得るために PE2 からのパーティション MDT を結合したため、( S1,G ) は PE2 からのパーティション MDT 上にも存在します。したがって PE3 は、結合した両方のパーティション MDT からストリーム ( S1、G ) を受信します。

PE3 は、( S1、G ) のパケットについて、PE1 から受信するものと PE2 から受信するものとを区別できません。どちらのストリームも、正しい RPF インターフェイスである、Lspvif0 で受信されます。

```
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

パケットが PE3 上の異なる入力物理インターフェイスに到着する場合と、同じインターフェイスに到着する場合があります。いずれの場合も、( S1、G ) の異なるストリームからのパケットは、PE3 で異なる MPLS ラベルで到着します。

```
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
```

## 解決策

解決策は、厳密な RPF を使用することです。厳密な RPF では、RPF インターフェイスで受信されるパケットがどのネイバーからのものがルータによりチェックされます。厳密な RPF がない場合には、入カインターフェイスが RPF インターフェイスかどうかというチェックのみ実行され、パケットがそのインターフェイス上の正しい RPF ネイバーから受信したものであるかどうかはチェックされません。

## Cisco IOS についての注

Cisco IOS での RPF についての重要な注意事項を次に示します。

- 厳密な RPF モードのオン/オフを切り替える際には、パーティション MDT の設定前にそれを設定するか、または BGP をクリアするかのいずれかにしてください。厳密な RPF コマンドだけを設定すると、別の Lspvif インターフェイスは直ちには作成されません。
- Cisco IOS において、厳密な RPF はデフォルトでは有効になっていません。
- デフォルト MDT プロファイルで **strict-rpf** コマンドを使用することはサポートされていません。

## 設定

Virtual Routing and Forwarding ( VRF ) については、PE3 上で厳密な RPF を設定できます。

```
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
!
```

RPF 情報が変更されました。

```
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 により、入力 PE ごとに Lspvif インターフェイスが作成されました。Lspvif インターフェイスは、入力 PE、アドレス ファミリ ( AF )、および VRF ごとに作成されます。10.100.1.6 の RPF が Lspvif1 を参照し、10.100.1.7 インターフェイスがインターフェイス Lspvif2 を参照するようになります。

```
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

ここで、RPF により、PE1 からのパケット ( S1、G ) が RPF インターフェイス Lspvif1 に対してチェックされます。これらのパケットは MPLS ラベル 29 で着信します。RPF により、PE2 からのパケット ( S2、G ) が RPF インターフェイス Lspvif2 に対してチェックされます。これらのパケットは MPLS ラベル 30 で着信します。異なる入力インターフェイスを通じてストリームが PE3 に到着しますが、これは同じインターフェイスの可能性もあります。ただし、mLDP が Penultimate Hop Popping ( PHP ) が使用することはないため、マルチキャスト パケットには常に通常の MPLS ラベルがあります。PE1 および PE2 から到着する ( S1、G ) パケットは 2 つの異なるパーティション MDT 上にあるため、それらの MPLS ラベルは異なります。したがって PE3 は、PE1 から着信する ( S1、G ) ストリームと、PE2 から着信する ( S1、G ) ストリームを区別できます。このようにして、パケットを PE3 と分離することができ、異なる複数の入力 PE ルータに対して RPF を実行することができます。

PE3 上の mLDP データベースには、入力 PE ごとに異なる Lspvif インターフェイスが示されるようになります。

```
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

```

マルチキャストストリームは入力 PE ごとに異なる MPLS ラベルで入力 PE に到達するため、厳密な RPF または入力 PE ごとの RPF が動作します。

```

PE3#show mpls forwarding-table vrf one
Local      Outgoing  Prefix          Bytes Label  Outgoing  Next Hop
Label      Label     or Tunnel Id    Switched     interface
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

```

厳密な RPF が動作している証拠として、PE3 上で転送される重複ストリーム (S1、G) がなくなっています。それでも PE3 には重複ストリームが到着しますが、RPF の障害のためそれはドロップされます。RPF 障害カウンタは 676255 であり、10 pps の一定速度で増えていきます。

```

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

```

この時点で PE3 での出力速度は 20 pps です。つまり、ストリーム (S1、G) と (S2、G) のそれぞれで 10 pps です。

```

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

```

## 結論

パーティション MDT を使用する mVPN 導入モデルでは、厳密な RPF チェックを使用する必要があります。

パーティション MDT を持つ mVPN 導入モデルに対して厳密な RPF チェックを設定しなくても、一見うまくいくように見えます。マルチキャスト ストリームはレシーバに配信されます。しかし、発信元が複数の入力 PE ルータに接続されている場合、重複したマルチキャストトラフィックが存在する可能性があります。これはネットワークにおける帯域幅の浪費につながり、レシーバでのマルチキャスト アプリケーションに悪影響を与える可能性があります。そのため、パーティション MDT を使用する mVPN 導入モデルでは、厳密な RPF チェックを設定することは必須です。